

INDEX

A

about windows. *See* **GnomeAbout**

abstract class. *See* class, abstract

accelerators, 182–183. *See also*

GtkAccelGroup

access methods (versus properties),
80–81

accessibility. *See* ATK

AC_CONFIG_FILES, 329–330

AC_CONFIG_HEADER, 328–329

AC_CONFIG_SRCDIR, 328–329

accumulators. *See* signals,

accumulators

AC_DEFINE_UNQUOTED, 329–330

AC_HEADER_STDC, 328–329

AC_INIT, 328–329

aclocal.m4, 327, 334

AC_OUTPUT, 329, 331

AC_PROG_CC, 328–329

AC_PROG_INTLTOOL, 329–330

AC_SUBST, 329–330

AH_TEMPLATE, 329–330

alignment containers. *See*

GtkAlignment

ALL_LINGUAS, 329–330

AM_GLIB_GNU_GETTEXT, 329–330

AM_INIT_AUTOMAKE, 328–329

AM_MAINTAINER_MODE, 328–329

application identifier, 247

application windows. *See* **GnomeApp**;

GtkWindow

arrays. *See* **GArray**

assertions, 27–28

assistants. *See* druids

asynchronous file access

canceling operations, 391, 396

description, 391–392

example, 396–408

file information, 394

opening and closing files,
392–393

reading and writing, 393

searching for directories,
394–395

symbolic links, 395–396

ATK, 112, 242, 321

atom, 9–10. *See also* memory chunks

attributes, 60

autoconf, 327, 334

autogen.sh, 336

autoheader, 327, 334

automake, 327, 334

Autotools. *See* GNU Autotools

B

base class

definition, 62

methods, 70–72

properties and, 71, 78

utility macros, 69–70

Battlestar Galactica, 122

binary data, 35

Bonobo, 246

Boolean values, 6

box containers. *See* **GtkBox**

Bradley, John, 111

Bugzilla, 430–432

button boxes

description, 139

example, 139–143

buttons. *See* **GtkButton**

C

C++ compatibility, 67
casting macro, 65–66
cell renderer. *See* **GtkCellRenderer**
check boxes. *See*
 GtkCellRendererToggle;
 GtkCheckButton;
 GtkMenuItem
child anchors. *See* **GtkTextBuffer**,
 child anchors
child properties, 145–146, 150
CLAMP(), 123, 138, 400
class
 abstract, 62, 68
 base (*see* base class)
 creating, 64–81
 definition, 60
 hierarchy, 62
 initializer, 68, 73–75, 77–78,
 85–87, 100–102
 properties (*see* properties)
 signals (*see* signals)
 structure, 64–65, 68, 72, 99
 type identifier (*see* type
 identifier)
 utility macros, 65–67, 100, 105
clipboard, 163, 239
color choosers. *See*
 GnomeColorPicker;
 GtkColorSelection
color, GDK, 294
combo box. *See* **GtkCombo**
compiling, difficulty with, 323–324
config.h.in, 327, 334
configuration systems, traditional,
 343–344
configure
 options, 335
 standard targets, 335
configure.ac (*configure.in*), 327–331
constructor, 60, 70–71
containers
 child, 112
 definition, 112
 value (*see* GValue)
 widgets (*see* **GtkContainer**;
 widget, containers)

context menus, 149, 269
controllers. *See* MVC
credits windows. *See* **GnomeAbout**
cursor, 232
CVS, 327

D

data pointer
 in objects, 133–136
 user (*see* user data pointer)
date and time widgets. *See*
 GnomeDateEdit
debugging using flow control, 27–28
decorations. *See* **GtkArrow**; **GtkFrame**
desktop files, 331–332, 336–337
destructor, 71
details. *See* signals, details
developer home, 432–433
dialogs. *See* **GnomeColorPicker**;
 GnomeFileEntry;
 GnomeFontPicker;
 GtkColorSelection;
 GtkDialog; **GtkFileSelection**;
 GtkFontSelection
directory operations, 389–391
DISTCLEANFILES, 331–332
dividers, 174–178. *See also* **GtkFrame**;
 GtkMenuItem; **GtkSeparator**
docked elements, 257
drop arrow, 191
druids, 286. *See also* **GnomeDruid**;
 GnomeDruidPageEdge;
 GnomeDruidPageStandard
dynamic module loader
 (GModule), 58

E

emission. *See* signals, emitting
emission hooks. *See* signals, hooks
entry box. *See* **GnomeEntry**; **GtkEntry**
error codes
 GConfClient, 364–365
 GnomeVFS, 383, 420–422
 Unicode, 22–23
error diagnosis. *See* GError; message
event-driven programming, 114

- events
 - definition, 112
 - signals and, 114, 117–118
- example
 - asynchronous file access, 396–408
 - attaching signal handlers, 93–95
 - button boxes, 139–143
 - child properties, 146
 - class initializer, 74–75, 77–78, 85–87, 102
 - class utility macros, 65–66, 100, 105
 - containers, 139–143
 - displaying help, 254
 - displaying URL, 253
 - GArray, 44–46
 - GConfClient** application, 368–374
 - GConfClient** data access, 354–355
 - GConfEntry, 360–362
 - GConfValue, 357–358
 - GdkPixbuf**, 132–137
 - GError, 29–32
 - getting source code, 4
 - GHashTable, 53–57
 - Glade-assisted program, 317–321
 - GList, 37–43
 - GNOME program skeleton, 248–249
 - GnomeAbout**, 284–285
 - GnomeApp**, 257–262
 - GnomeClient**, 300–301, 404–406
 - GnomeColorPicker**, 270–275
 - GnomeDateEdit**, 270–275
 - GnomeDruid**, 286–292
 - GnomeDruidPageEdge**, 286–292
 - GnomeDruidPageStandard**, 286–292
 - GnomeEntry**, 270–275
 - GnomeFileEntry**, 270–275
 - GnomeFontPicker**, 270–275
 - GnomeHRef**, 281–282
 - GnomeIconEntry**, 270–275
 - GnomePixmapEntry**, 270–275
 - GnomeVFS file information, 422–427
 - GNU Autotools, 328–341
 - GString, 34–36
 - GTimer, 23
 - GTK+, 114–118
 - GtkAdjustment**, 157–161
 - GtkArrow**, 175–177
 - GtkBox**, 139–143
 - GtkButton**, 151–155
 - GtkCellRenderer**, 192–195, 215
 - GtkCheckButton**, 151–155
 - GtkColorSelection**, 170–174
 - GtkCombo**, 157–161
 - GtkDialog**, 186–189, 286–287
 - GtkEntry**, 157–161
 - GtkFileSelection**, 170–174
 - GtkFontSelection**, 170–174
 - GtkFrame**, 175–177
 - GtkImage**, 122–126, 132–137
 - GtkLabel**, 122–126
 - GtkListStore**, 192–195
 - GtkMenu**, 257–262
 - GtkMenuBar**, 257–262
 - GtkMenuItem**, 257–262
 - GtkMenuShell**, 257–262
 - GtkNotebook**, 139–143
 - GtkOptionMenu**, 157–161
 - GtkPaned**, 139–143
 - GtkProgressBar**, 122–126
 - GtkRadioButton**, 151–155
 - GtkRadioMenuItem**, 264
 - GtkScale**, 132–137, 157–161
 - GtkScrolledWindow**, 184–185
 - GtkSeparator**, 175–177
 - GtkSpinButton**, 157–161
 - GtkStatusbar**, 257–262
 - GtkTable**, 139–143
 - GtkTextBuffer**, 217–224
 - GtkTextIter, 217–224
 - GtkTextView**, 217–224
 - GtkToolbar**, 257–262
 - GtkTooltips**, 179–181
 - GtkTreeIter, 210–217
 - GtkTreeStore**, 210–217
 - GtkTreeView**, 192–195, 210–217
 - GtkTreeViewColumn**, 192–195
 - GtkWindow**, 114–118
 - GTree, 47–51
 - interface base initializer, 106

- example, *continued*
 - interface initializer, 107–108
 - interface method, 108
 - locating files, 250–251
 - Makefile.am*, 331–333
 - marshallers, 90–92
 - memory chunks, 9–11
 - message logging, 24–25
 - MIME types, 415–416
 - print handler, 26
 - quarks, 12
 - references, 83–84
 - signal definition, 85–87
 - signal details, 95
 - signal handlers, 87–88, 93–94
 - signal hooks, 96–97
 - signal listing and printing, 98–99
 - structure, 4
 - synchronous file access, 381–383
 - temperature converter, 303–321
 - type identifier function, 67, 100–101, 107
 - using objects, 82–84, 103–104
- exception handling. *See* `GError`
- executing programs, 251–253
- `EXTRA_DIST`, 331–332

F

- fatal errors, 24–25
- file browsers. *See* `GnomeFileEntry`;
`GtkFileSelection`
- files
 - accessing (*see* `GnomeVFS`)
 - locating, 249–251
 - management (*see* synchronous file access, file management)
- finalizer, 71–72
- focus. *See* input focus
- font choosers. *See* `GnomeFontPicker`;
`GtkFontSelection`
- fonts, 127–128
- functions and macros, difference between, 4

G

- `GArray`, 44–46. *See also* `GList`
- `g_array_append_vals()`, 45

- `g_array_free()`, 46
- `g_array_index()`, 44–45
- `g_array_insert_vals()`, 45
- `g_array_new()`, 44
- `g_array_prepend_val()`, 45
- `g_array_prepend_vals()`, 45
- `g_array_remove_index()`, 46
- `g_array_remove_index_fast()`, 46
- `g_array_set_size()`, 45
- `g_array_sized_new()`, 44
- `g_array_sort()`, 46
- `g_array_sort_with_data()`, 46
- `g_ascii_dtostr()`, 15
- `g_ascii_strcasecmp()`, 12
- `g_ascii_strtod()`, 15
- `g_assert()`, 28
- `g_assert_not_reached()`, 28
- `GBaseFinalizeFunc`, 68–69
- `GBaseInitFunc`, 68–69
- `g_blow_chunks()`, 10
- `GBoxed`, 109
- `GByteArray`, 57
- `g_cclosure_marshal_*()`, 89–90
- `GClassFinalizeFunc`, 68–69
- `GClassInitFunc`, 68–69
- `g_clear_error()`, 29–30
- `GClosure`, 91, 109
- `GCompareFunc`, 42–43
- `GConf`
 - configuration editor, 345–346
 - data types, 345
 - keys, 345
 - overview, 344–345
 - path, 344–345, 351–352
 - programming with (*see* `GConfClient`)
 - schemas, 345–346, 366–368, 374
 - user programs, 345–350
- `GConfClient`
 - accessing directories, 359–360
 - accessing keys and values, 359–360 (*see also* `GConfEntry`)
 - accessing values, 352–358 (*see also* `GConfValue`)
 - application skeleton, 351–352
 - cache, 364
 - error handling, 364–366
 - example, 354–355

GConfClient, *continued*

- example application, 368–374
- initializing, 350–352
- value change notification, 362–364
- `gconf_client_add_dir()`, 351–352
- `gconf_client_all_dirs()`, 359
- `gconf_client_all_entries()`, 359–360
- `gconf_client_clear_cache()`, 364
- `gconf_client_dir_exists()`, 359–360
- `gconf_client_get()`, 356
- `gconf_client_get_bool()`, 353, 355
- `gconf_client_get_default()`, 351–352
- `gconf_client_get_entry()`, 359
- `gconf_client_get_float()`, 352
- `gconf_client_get_int()`, 352, 354
- `gconf_client_get_list()`, 353–354
- `gconf_client_get_pair()`, 353
- `gconf_client_get_string()`, 353, 355
- `gconf_client_key_is_writable()`, 354
- `gconf_client_notify_add()`, 363
- `GConfClientNotifyFunc`, 363
- `gconf_client_notify_remove()`, 363–364
- `gconf_client_preload()`, 364
- `gconf_client_remove_dir()`, 351–352
- `gconf_client_set()`, 356
- `gconf_client_set_bool()`, 353, 355
- `gconf_client_set_error_handling()`, 352, 362, 365
- `gconf_client_set_float()`, 353, 355
- `gconf_client_set_int()`, 353, 355
- `gconf_client_set_list()`, 354–355
- `gconf_client_set_pair()`, 354–355
- `gconf_client_set_string()`, 353, 355
- `gconf_client_suggest_sync()`, 364
- `gconf_client_unset()`, 360
- `GConfEntry`, 359–362
- `gconf_entry_free()`, 359, 362
- `gconf_entry_get_key()`, 359, 362
- `gconf_entry_get_value()`, 359, 361
- `gconf_entry_steal_value()`, 359
- `gconfntool-2`, 347–350
- `GConfValue`
 - example, 357–358
 - operation, 355–356
- `gconf_value_copy()`, 357
- `gconf_value_free()`, 356, 358
- `gconf_value_get_bool()`, 356
- `gconf_value_get_car()`, 357
- `gconf_value_get_cdr()`, 357
- `gconf_value_get_float()`, 356
- `gconf_value_get_int()`, 356, 358
- `gconf_value_get_list()`, 357
- `gconf_value_get_string()`, 356
- `gconf_value_new()`, 356–358
- `gconf_value_set_bool()`, 356
- `gconf_value_set_car()`, 357
- `gconf_value_set_car_nocopy()`, 357
- `gconf_value_set_cdr()`, 357
- `gconf_value_set_cdr_nocopy()`, 357
- `gconf_value_set_float()`, 356
- `gconf_value_set_int()`, 356–357
- `gconf_value_set_list()`, 357
- `gconf_value_set_list_nocopy()`, 357–358
- `gconf_value_set_list_type()`, 357–358
- `gconf_value_set_string()`, 356, 358
- `gconf_value_to_string()`, 357–358, 361
- `g_create_hash_table_new()`, 52–53
- `g_critical()`, 24–25
- `GDestroyNotify`, 47
- `g_direct_equal()`, 73
- `g_direct_hash()`, 53
- `GDK`, 111, 130–132, 242
- `gdk_color_parse()`, 289, 291, 294
- GdkPixbuf**, 130–137
- `gdk_pixbuf_add_alpha()`, 131
- GdkPixbufAnimation**, 243
- `gdk_pixbuf_composite()`, 131
- `gdk_pixbuf_copy()`, 130
- `gdk_pixbuf_copy_area()`, 130
- `gdk_pixbuf_fill()`, 132
- `gdk_pixbuf_get_has_alpha()`, 131
- `gdk_pixbuf_get_height()`, 130
- `gdk_pixbuf_get_width()`, 130
- GdkPixbufLoader**, 243
- `gdk_pixbuf_new_from_file()`, 130
- `gdk_pixbuf_saturate_and_pixelate()`, 131–132
- `gdk_pixbuf_scale()`, 131
- `gdk_pixbuf_scale_simple()`, 131, 134
- `GEqualFunc`, 52–53
- `GError`
 - defining error domains and codes, 30–32
 - description, 28
 - example, 29–32

GError, *continued*

- setting error conditions, 31–32
- structure, 28
- `g_error_free()`, 29–30
- `g_error_matches()`, 29–30
- `gettext`, 327, 329–330, 333, 339–341
- `GETTEXT_PACKAGE`, 329–330
- `g_extension_pointer()`, 251
- `g_filename_from_uri()`, 22
- `g_filename_from_utf8()`, 22
- `g_filename_to_uri()`, 22
- `g_filename_to_utf8()`, 22
- `g_free()`, 7–8
- `g_free_list()`, 43
- GFunc, 41
- GHashFunc, 52–53
- GHashTable
 - creating, 52–53
 - deleting, 57
 - deleting entries, 56
 - description, 52
 - example, 53–57
 - finding size, 54
 - hash function, 52–53
 - hash value, 52–53
 - inserting and replacing entries, 54
 - iterating through, 56–57
 - lookups, 55
- `g_hash_table_destroy()`, 57
- `g_hash_table_foreach()`, 56–57
- `g_hash_table_foreach_remove()`, 56
- `g_hash_table_foreach_steal()`, 56
- `g_hash_table_insert()`, 54
- `g_hash_table_lookup()`, 55
- `g_hash_table_lookup_extended()`, 55
- `g_hash_table_new_full()`, 53
- `g_hash_table_remove()`, 56
- `g_hash_table_replace()`, 54
- `g_hash_table_size()`, 54
- `g_hash_table_steal()`, 56
- GHFunc and GHRFunc, 56
- `g_int_equal()`, 53
- GInterfaceFinalizeFunc, 107
- GInterfaceInitFunc, 107
- `g_int_hash()`, 53
- `G_IS_OBJECT()`, 70
- `G_IS_OBJECT_CLASS()`, 70

Glade

- clipboard, 311
- creating a project, 304
- description, 303
- example use of libglade, 317–321
- interface file (`.glade`), 312–315
- making room for widgets in
 - containers, 309–310
- property editor, 306–310
- signals, 316–317
- starting, 303–304
- user interface integration into
 - applications, 315–317
- widget palette, 305
- window titles, 308
- `glade_get_widget_name()`, 317
- `glade_get_widget_tree()`, 317
- `glade_xml_get_widget()`, 316, 320
- `glade_xml_get_widget_prefix()`, 316
- `glade_xml_new()`, 315, 320
- `glade_xml_signal_autoconnect()`, 317
- `glade_xml_signal_connect()`, 317
- `glade_xml_signal_connect_data()`, 317, 321

GLib

- naming conventions (*see* naming conventions)
- overview, 5
- types, 6–7

glib-genmarshal, 90–92

GList

- adding elements, 37–38
- concatenating, 43
- copying, 43
- creating, 36–37
- deallocating, 43
- example, 37–43
- iterating through, 41–43
- navigating, 38–40
- removing elements, 40–41
- reversing, 43
- sorting, 42–43
- `g_list_append()`, 37
- `g_list_concat()`, 43
- `g_list_copy()`, 43
- `g_list_find()`, 38
- `g_list_first()`, 39
- `g_list_foreach()`, 41–42

- `g_list_index()`, 39–40
- `g_list_insert()`, 37–38
- `g_list_insert_before()`, 38
- `g_list_last()`, 39
- `g_list_length()`, 39–40
- `g_list_next()`, 39–40
- `g_list_nth()`, 39
- `g_list_nth_data()`, 39
- `g_list_nth_prev()`, 39
- `g_list_position()`, 39–40
- `g_list_prepend()`, 37, 40
- `g_list_prev()`, 39–40
- `g_list_remove()`, 40
- `g_list_remove_all()`, 40
- `g_list_reverse()`, 43
- `g_list_sort()`, 42
- `g_list_sort_with_data()`, 43
- `g_locale_from_utf8()`, 22
- `g_locale_to_utf8()`, 22
- `G_LOG_DOMAIN`, 24
- `g_log_set_always_fatal()`, 25
- `g_log_set_handler()`, 57
- `g_malloc()`, 7–8
- `g_malloc0()`, 7–8
- GMemChunk. *See* memory chunks
- `g_mem_chunk_alloc()`, 9–11
- `g_mem_chunk_alloc0()`, 10
- `g_mem_chunk_clean()`, 10
- `g_mem_chunk_create()`, 11
- `g_mem_chunk_destroy()`, 10
- `g_mem_chunk_free()`, 10
- `g_mem_chunk_info()`, 11
- `g_mem_chunk_new()`, 9–11
- `g_mem_chunk_new0()`, 11
- `g_mem_chunk_print()`, 11
- `g_message()`, 24–26
- `g_new()`, 8–9
- `g_new0()`, 8
- GNode, 58
- GNOME
 - application menu files
(*see* desktop files)
 - file domain, 249–250
 - Foundation, 430
 - future of, 434–436
 - learning about, 429
 - libraries, 245–246
 - license, 2

- opposition to, 246
- software downloads, 430
- Usability Project, 432 (*see also* user interface guidelines)
- utility functions, 249–254
- widgets in Glade, 305
- GNOME application
 - compiling, 249, 324–326
 - initializing, 247–248
 - windows (*see* **GnomeApp**)
- GnomeAbout**
 - example, 284–285
 - operation, 283–284
- `gnome_about_new()`, 283–285
- GnomeApp**
 - description, 257
 - example, 257–262
 - titles, 257
- GnomeAppBar**, 302
- `gnome-app-helper`, 302
- `gnome_app_set_contents()`, 261
- `gnome_app_set_menus()`, 261, 266
- `gnome_app_set_statusbar()`, 261, 268
- `gnome_app_set_toolbar()`, 261, 268
- GnomeClient**
 - example, 300–301, 404–406
 - operation, 298–300
- `GNOME_CLIENT_CONNECTED()`, 298–299
- `gnome_client_save_any_dialog()`, 300
- `gnome_client_save_error_dialog()`, 300
- `gnome_client_set_clone_command()`, 300–301
- `gnome_client_set_current_directory()`, 300–301
- `gnome_client_set_discard_command()`, 300
- `gnome_client_set_environment()`, 300
- `gnome_client_set_restart_command()`, 300–301
- GnomeColorPicker**
 - example, 270–275
 - operation, 278 (*see also* **GtkColorSelection**)
- GnomeDateEdit**
 - example, 270–275
 - future of, 435
 - operation, 280–281
- GnomeDruid**
 - description, 286

GnomeDruid, *continued*
 example, 286–292
 operation, 296–297
 gnome_druid_append_page(), 291, 296
 gnome_druid_insert_page(), 297
 gnome_druid_new_with_window(), 291, 296
GnomeDruidPage. *See* **GnomeDruid**,
 description

GnomeDruidPageEdge
 description, 292
 example, 286–292
 operation, 292–294
 gnome_druid_page_edge_new_with_vals(),
 289, 292–293
 gnome_druid_page_edge_set_bg_color(),
 289, 291, 293
 gnome_druid_page_edge_set_logo_bg_
 color(), 294
 gnome_druid_page_edge_set_textbox_
 color(), 291, 294
 gnome_druid_page_edge_set_text_color(),
 291, 294
 gnome_druid_page_edge_set_title_
 color(), 289, 291, 294

GnomeDruidPageStandard
 description, 292
 example, 286–292
 operation, 295–296
 gnome_druid_page_standard_append_
 item(), 290–291, 296
 gnome_druid_prepend_page(), 297
 gnome_druid_set_page(), 297

GnomeEntry
 example, 270–275
 operation, 275–276 (*see also*
GtkCombo; **GtkEntry**)
 gnome_entry_append_history(), 275
 gnome_entry_clear_history(), 275
 gnome_entry_get_max_saved(), 276
 gnome_entry_prepend_history(), 275
 gnome_entry_set_max_saved(), 276
 gnome_execute_async(), 252
 gnome_execute_async_fds(), 252
 gnome_execute_async_with_env(), 252
 gnome_execute_async_with_env_fds(), 252
 gnome_execute_shell(), 252
 gnome_execute_shell_fds(), 253
 gnome_execute_terminal_shell(), 253
 gnome_execute_terminal_shell_fds(), 253

GnomeFileEntry
 example, 270–275
 operation, 276–277 (*see also*
GtkFileSelection)
 gnome_file_entry_get_full_path(), 271,
 276–277

GnomeFontPicker
 example, 270–275
 operation, 277–278 (*see also*
GtkFontSelection)
 gnome_font_picker_uw_set_widget(),
 273, 277
 gnome_gconf_get_app_settings_
 relative(), 351–352
 gnome_gtk_widget_add_popup_items(), 269
 gnome_help_display(), 254

GnomeHRef, 281–282

GnomeIconEntry
 example, 270–275
 operation, 278–279

GnomeIconList, 302

GnomeIconSelection, 302
 gnome_master_client(), 298, 301

GnomePixmapEntry
 example, 270–275
 operation, 279–280
 gnome_pixmap_entry_get_filename(),
 272, 280
 gnome_pixmap_entry_new(), 274, 279
 gnome_pixmap_entry_set_pixmap_subdir(),
 280
 gnome_pixmap_entry_set_preview_size(),
 280
 gnome_popup_menu_append(), 269
 gnome_popup_menu_attach(), 269
 gnome_popup_menu_get_accel_group(), 269
 gnome_popup_menu_new(), 269
 gnome_popup_menu_new_with_accelgroup(),
 269
 gnome_prepend_terminal_to_vector(), 252

GnomeProgram, 247
 gnome_program_init(), 247–248, 259
 gnome_program_locate_file(), 249–251
 gnome_score_get_notable(), 256
 gnome_score_init(), 256, 283
 gnome_score_log(), 256

GnomeScores, 282–283

- gnome_scores_display_with_pixmap(), 282–283
- gnome_sound_play(), 254
- GnomeThumbnail**, 302
- gnome_triggers_do(), 255–256
- gnome_url_show(), 253
- gnome_url_show_with_env(), 253
- gnome_util_home_file(), 251
- gnome_util_prepend_user_home(), 251
- GnomeVFS**
 - asynchronous access
 - (*see* asynchronous file access)
 - directory operations
 - (*see* directory operations)
 - error codes (*see* error codes, **GnomeVFS**)
 - executable information, 420
 - file access types, 378
 - file information example, 422–427
 - file permissions, 384
 - free space, 420
 - icons, 420
 - initializing and shutting down, 379–380
 - local file access, 420
 - MIME types (*see* MIME types)
 - portability notes, 422
 - purpose, 377
 - synchronous access
 - (*see* synchronous file access)
 - transfers, 409–413
 - URI, 377–379
 - URI structures (*see* **GnomeVFSURI**)
- GnomeVFSAsyncCallback**, 393
- gnome_vfs_async_cancel(), 391, 396, 399
- gnome_vfs_async_close(), 393, 399–401
- GnomeVFSAsyncCloseCallback**, 393, 399
- gnome_vfs_async_create(), 392
- gnome_vfs_async_create_symbolic_link(), 395–396
- gnome_vfs_async_create_uri(), 393
- GnomeVFSAsyncDirectoryLoadCallback**, 396
- gnome_vfs_async_find_directory(), 395
- GnomeVFSAsyncFindDirectoryCallback**, 396
- gnome_vfs_async_get_file_info(), 394
- GnomeVFSAsyncGetFileInfoCallback**, 394
- GnomeVFSAsyncHandle**, 391
- gnome_vfs_async_load_directory()**, 394–395
- gnome_vfs_async_load_directory_uri()**, 395
- gnome_vfs_async_open(), 392, 403
- GnomeVFSAsyncOpenCallback**, 393, 401
- gnome_vfs_async_open_uri(), 392
- gnome_vfs_async_read(), 393, 401–402
- GnomeVFSAsyncReadCallback**, 393, 400
- gnome_vfs_async_set_file_info(), 394
- GnomeVFSAsyncSetFileInfoCallback**, 394
- gnome_vfs_async_write(), 393
- GnomeVFSAsyncWriteCallback**, 393
- gnome_vfs_async_xfer(), 412–413
- GnomeVFSAsyncXferProgressCallback**, 413
- gnome_vfs_check_same_fs(), 388
- gnome_vfs_close(), 382, 384
- gnome_vfs_create(), 382–383
- gnome_vfs_create_symbolic_link(), 389
- gnome_vfs_create_uri(), 384
- gnome_vfs_directory_close(), 390
- gnome_vfs_directory_list_load(), 390
- gnome_vfs_directory_open(), 389
- gnome_vfs_directory_open_from_uri(), 389
- gnome_vfs_directory_read_next(), 390
- gnome_vfs_directory_visit(), 390
- gnome_vfs_directory_visit_files(), 390
- gnome_vfs_directory_visit_files_at_uri(), 390
- GnomeVFSDirectoryVisitFunc**, 390–391
- gnome_vfs_directory_visit_uri(), 390
- GnomeVFSFileInfo**, 385–388
- gnome_vfs_file_info_clear(), 388
- gnome_vfs_file_info_copy(), 388
- gnome_vfs_file_info_dup(), 388
- gnome_vfs_file_info_list_free(), 390
- gnome_vfs_file_info_matches(), 388
- gnome_vfs_file_info_new(), 388, 402
- gnome_vfs_file_info_ref(), 388
- gnome_vfs_file_info_unref(), 388, 403
- gnome_vfs_find_directory(), 391
- GnomeVFSFindDirectoryResult**, 395
- gnome_vfs_format_file_size_for_display(), 422
- gnome_vfs_get_file_info(), 385, 402
- gnome_vfs_get_file_info_from_handle(), 388

GnomeVFSGetFileInfoResult, 394
 gnome_vfs_get_file_info_uri(), 387
 gnome_vfs_get_full_path(), 402
 gnome_vfs_get_local_path_from_uri(),
 420
 gnome_vfs_get_uri_from_local_path(),
 382, 402, 420
 gnome_vfs_get_volume_free_space(), 420
 GnomeVFSHandle, 383
 gnome_vfs_icon_path_from_filename(),
 420
 gnome_vfs_init(), 379–380
 gnome_vfs_initialized(), 380
 gnome_vfs_is_executable_command_
 string(), 420
 gnome_vfs_make_directory(), 389
 gnome_vfs_make_directory_for_uri(), 389
 GnomeVFSMimeTypeApplication, 414
 gnome_vfs_mime_application_copy(), 414
 gnome_vfs_mime_application_free(), 414
 gnome_vfs_mime_application_list_free(),
 414
 gnome_vfs_mime_can_be_executable(), 414
 gnome_vfs_mime_get_all_applications(),
 413
 gnome_vfs_mime_get_default_action_
 type(), 413
 gnome_vfs_mime_get_default_
 application(), 413
 gnome_vfs_mime_get_description(), 414
 gnome_vfs_mime_get_icon(), 413
 gnome_vfs_mime_get_short_list_
 applications(), 413
 gnome_vfs_mime_id_in_application_
 list(), 414
 gnome_vfs_mime_type_is_known(), 413
 gnome_vfs_move(), 388
 gnome_vfs_move_uri(), 388
 gnome_vfs_open(), 383
 gnome_vfs_open_uri(), 383
 gnome_vfs_read(), 385
 gnome_vfs_remove_directory(), 389
 gnome_vfs_remove_directory_from_uri(),
 389
 GnomeVFSResult, 420
 gnome_vfs_result_to_string(), 421–422
 gnome_vfs_seek(), 385
 gnome_vfs_set_file_info(), 387
 gnome_vfs_set_file_info_uri(), 388
 gnome_vfs_shutdown(), 379–380
 gnome_vfs_tell(), 385
 gnome_vfs_truncate(), 389
 gnome_vfs_truncate_handle(), 389
 gnome_vfs_truncate_uri(), 389
 gnome_vfs_unlink(), 388
 gnome_vfs_unlink_from_uri(), 388
 GnomeVFSURI
 building, 417
 connection information, 419
 description, 417
 extracting path information,
 418–419
 lists of, 419
 gnome_vfs_uri_append_file_name(), 417
 gnome_vfs_uri_append_path(), 417
 gnome_vfs_uri_append_string(), 417
 gnome_vfs_uri_dup(), 417
 gnome_vfs_uri_exists(), 418
 gnome_vfs_uri_extract_dirname(), 418
 gnome_vfs_uri_extract_short_name(),
 401, 419
 gnome_vfs_uri_extract_short_path_
 name(), 419
 gnome_vfs_uri_get_fragment_
 identifier(), 418
 gnome_vfs_uri_get_host_name(), 419
 gnome_vfs_uri_get_host_port(), 419
 gnome_vfs_uri_get_parent(), 418
 gnome_vfs_uri_get_password(), 419
 gnome_vfs_uri_get_path(), 418
 gnome_vfs_uri_get_scheme(), 418
 gnome_vfs_uri_get_user_name(), 419
 gnome_vfs_uri_has_parent(), 418
 gnome_vfs_uri_is_local(), 418
 gnome_vfs_uri_is_parent(), 418
 gnome_vfs_uri_list_copy(), 419
 gnome_vfs_uri_list_free(), 419
 gnome_vfs_uri_list_ref(), 419
 gnome_vfs_uri_list_unref(), 419
 gnome_vfs_uri_new(), 401, 417
 gnome_vfs_uri_ref(), 417
 gnome_vfs_uri_set_host_name(), 419
 gnome_vfs_uri_set_host_port(), 419
 gnome_vfs_uri_set_password(), 419
 gnome_vfs_uri_set_user_name(), 419
 gnome_vfs_uri_to_string(), 418

- gnome_vfs_uri_unref(), 401, 417
- gnome_vfs_write(), 382, 385
- gnome_vfs_xfer_delete_list(), 412
- GnomeVFSXferProgressCallback, 410
- gnome_vfs_xfer_uri(), 409–410
- gnome_vfs_xfer_uri_list(), 412
- gnome_window_toplevel_set_title(), 257
- GNU Autotools
 - description, 326–327
 - example, 328–341
 - outfitting source code with, 327–328
- GObject, 60
- G_OBJECT(), 70
- g_object_add_weak_pointer(), 83–84
- G_OBJECT_CLASS(), 70
- g_object_class_install_property(), 77–78
- g_object_get(), 82–83
- g_object_get_data(), 133
- g_object_get_property(), 83
- g_object_get_valist(), 83
- g_object_new(), 81–82
- g_object_ref(), 83–84
- g_object_remove_weak_pointer(), 83–84
- g_object_set(), 82
- g_object_set_data(), 135–136
- g_object_set_property(), 83
- g_object_set_valist(), 83
- g_object_unref(), 83–84
- GParamSpec, 73–75
- g_param_spec_*(), 73–74
- g_print(), 25–26
- g_printerr(), 25–26
- g_printf_string_upper_bound(), 15
- GPrintFunc, 27
- g_propagate_error(), 32
- GPtrArray, 57
- g_qsort_with_data(), 57
- GQuark. *See* quarks
- g_quark_from_static_string(), 12
- g_quark_from_string(), 12
- g_quark_to_string(), 12
- g_quark_try_string(), 12
- GQueue, 58
- graphical user interfaces,
 - programming. *See* toolkits
- gravity, 232–234
- g_realloc(), 8
- g_renew(), 8
- g_return_if_fail(), 28
- g_return_if_reached(), 27
- g_return_val_if_fail(), 28
- g_set_error(), 31–32
- g_set_printerr_handler(), 26–27
- g_set_print_handler(), 26–27
- GSignalAccumulator, 92
- g_signal_add_emission_hook(), 97
- g_signal_connect(), 93–94
- g_signal_connect_after(), 94
- g_signal_connect_data(), 99
- g_signal_connect_swapped(), 94
- GSignalEmissionHook, 96
- g_signal_emit(), 88–89
- g_signal_emit_by_name(), 88–89
- g_signal_emitv(), 89
- g_signal_emit_valist(), 89
- g_signal_handler_block(), 98
- g_signal_handler_disconnect(), 94
- g_signal_handler_find(), 99
- g_signal_handler_is_connected(), 94–95
- g_signal_handlers_block_by_func(), 99
- g_signal_handlers_block_matched(), 99
- g_signal_handlers_disconnect_by_func(), 99
- g_signal_handlers_disconnect_matched(), 99
- g_signal_handlers_unblock_by_func(), 99
- g_signal_handlers_unblock_matched(), 99
- g_signal_handler_unblock(), 98
- g_signal_hook_remove(), 97
- GSignalInvocationHint, 92–93
- g_signal_list_ids(), 98
- g_signal_lookup(), 98
- g_signal_name(), 98
- g_signal_new(), 86–87
- g_signal_neww(), 99
- g_signal_query(), 99
- g_signal_stop_emission(), 98
- g_signal_stop_emission_by_name(), 98
- g_signal_valist(), 99
- GSList, 57
- g_snprintf(), 14
- GSpawn, 252
- g_stpcpy(), 14
- g_strcanon(), 14

- g_strchomp(), 14
- g_strchug(), 14
- g_strcompress(), 13
- g_strconcat(), 13
- g_strdelimit(), 14
- g_strdup(), 13
- g_strdup_printf(), 13
- g_strdupv(), 15
- g_strdup_vprintf(), 13
- g_str_equal(), 53
- g_strerror(), 27
- g_strescape(), 13
- g_strfreev(), 15
- g_str_hash(), 53
- GString
 - adding, removing characters and data, 34–35
 - comparing, 35–36
 - creating, 34–35
 - deallocating, 36
 - description, 33
 - example, 34–36
 - typical operation, 33–34
- g_string_append(), 34
- g_string_append_c(), 34
- g_string_append_len(), 35
- g_string_append_printf(), 36
- g_string_append_unichar(), 34
- g_string_assign(), 33–34
- GStringChunk, 57
- g_string_equal(), 35
- g_string_erase(), 35
- g_string_free(), 36
- g_string_hash(), 36
- g_string_insert(), 34–35
- g_string_insert_c(), 34
- g_string_insert_len(), 35
- g_string_insert_unichar(), 34
- g_string_new(), 33
- g_string_new_len(), 34
- g_string_prepend*(), 34
- g_string_prepend_len(), 35
- g_string_printf(), 36
- g_string_set_size(), 35
- g_string_sized_new(), 33–34
- g_string_truncate(), 35
- g_strjoin(), 13
- g_str_joinv(), 15
- g_strndup(), 13
- g_strreverse(), 14
- g_strrstr(), 14
- g_strrstr_len(), 15
- g_strsignal(), 27
- g_strsplit(), 15
- g_strstrip(), 14
- g_strstr_len(), 14
- g_timeout_add(), 224
- GTimer, 23–24
- g_timer_*(), 23–24
- GTK+
 - compiling programs, 116, 324–326
 - components, 111–112
 - example, 114–118
 - program behavior, 116
 - program structure, 117
 - terminating programs, 117–118
 - themes, 113–114, 244
 - widgets in Glade, 305
- gtk_accelerator_valid(), 182
- GtkAccelGroup**, 182, 259, 263, 269
- gtk_accel_groups_from_object(), 182
- GtkAdjustment**
 - example, 157–161
 - operation, 165–166
- gtk_adjustment_changed(), 165
- gtk_adjustment_get_value(), 165
- gtk_adjustment_new(), 158, 165
- gtk_adjustment_set_value(), 165
- GtkAlignment**
 - description, 139
 - operation, 150
- GtkArrow**
 - description, 175
 - example, 175–177
 - operation, 178
- GtkAspectFrame**, 243
- GtkBox**
 - child properties, 145–146
 - creating with Glade, 309–310
 - description, 139
 - example, 139–143
 - properties, 145
- gtk_box_pack_end(), 144–145
- gtk_box_pack_end_defaults(), 145
- gtk_box_pack_start(), 144–145

`gtk_box_pack_start_defaults()`, 141, 145

GtkButton

description, 151
example, 151–155
properties and signals, 155–156

GtkCalendar, 243. *See also*

GnomeDataEdit

`GtkCallback`, 144

GtkCellRenderer

description, 192
example, 192–195, 215
operation, 206–207

GtkCellRendererPixbuf, 206–207

GtkCellRendererText, 206–207

`gtk_cell_renderer_text_new()`, 193,
195, 215

GtkCellRendererToggle, 206–207

`gtk_cell_renderer_toggle_new()`, 215

GtkCheckButton

description, 151
example, 151–155
properties and signals, 155–156

GtkCheckMenuItem. *See*

GtkMenuItem

`gtk_check_menu_item_new_with_label()`,
264

`gtk_check_menu_item_new_with_
mnemonic()`, 263

GtkClipboard, 239

`gtk_clipboard_get()`, 239

GtkColorSelection

description, 161
example, 170–174
operation, 168

GtkColorSelectionDialog.

See **GtkColorSelection**

`gtk_color_selection_is_adjusting()`,
168, 171

GtkCombo

description, 156
example, 157–161
operation, 164–165

`gtk_combo_set_popdown_strings()`,
159, 164

GtkContainer, 143–144

`gtk_container_add()`, 116–117, 140,
143–144

`gtk_container_child_get_property()`, 146

`gtk_container_child_set_property()`, 146

`gtk_container_foreach()`, 144

`gtk_container_remove()`, 143

GtkCurve, 243

GtkDialog

creating with Glade, 305

description, 186

example, 186–189, 286–287

operation, 189–191

`gtk_dialog_new_with_buttons()`, 187–190

`gtk_dialog_set_default_response()`,
188, 190

GtkDragDestIface, 202

GtkDragSourceIface, 202

GtkEditable (interface), 163–164

`gtk_editable_copy_clipboard()`, 163

`gtk_editable_cut_clipboard()`, 163

`gtk_editable_delete_clipboard()`, 163

`gtk_editable_delete_text()`, 163

`gtk_editable_get_chars()`, 163

`gtk_editable_insert_text()`, 163

`gtk_editable_paste_clipboard()`, 163

GtkEntry

description, 156

example, 157–161

methods, 163–164

properties, 162–163

signals, 163

GtkEventBox, 242

GtkFileSelection

description, 162

example, 170–174

operation, 168–169

`gtk_file_selection_get_selections()`,
170

GtkFixed, 243

GtkFontSelection

description, 161

example, 170–174

operation, 168–169

GtkFontSelectionDialog. *See*

GtkFontSelection

GtkFrame

description, 174

example, 175–177

operation, 177–178

GtkGammaCurve, 243

GtkHBox. *See* button boxes; **GtkBox**

GtkHPaned. *See* **GtkPaned**

GtkHScale. *See* **GtkScale**

GtkHScrollBar. *See* **GtkScrollBar**

GtkHSeparator. *See* **GtkSeparator**

GtkImage

example, 122–126, 132–137

manipulation (*see* image
manipulation)

memory leaks, 134

properties, 129–130

GtkImageMenuItem. *See*

GtkMenuItem

`gtk_image_menu_item_new_from_stock()`,
259–260, 263

`gtk_image_menu_item_new_with_label()`,
264

`gtk_image_menu_item_new_with_
mnemonic()`, 263

`gtk_init()`, 115

GtkLabel

creating with Glade, 308–309

example, 122–126

markup language (*see* Pango,
markup language)

properties, 128–129

GtkListStore

accessing nodes, 196–198

creating, 196

description, 192

example, 192–195

iterating through, 198

removing nodes, 197

setting and adding nodes,
196–197

`gtk_list_store_append()`, 193, 195–196,
197

`gtk_list_store_clear()`, 197

`gtk_list_store_insert()`, 197

`gtk_list_store_insert_after()`, 197

`gtk_list_store_insert_before()`, 197

`gtk_list_store_new()`, 193, 195–196

`gtk_list_store_prepend()`, 197

`gtk_list_store_remove()`, 197

`gtk_list_store_set()`, 193, 195–196

`gtk_list_store_set_value()`, 196

`gtk_main()`, 116–117

`gtk_main_quit()`, 115, 117

GtkMenu

creating and installing
(summary), 265–266

description, 262

example, 257–262

GtkMenuBar

description, 262

example, 257–262

operation, 265–266

GtkMenuItem

description, 262

example, 257–262

example of **GtkRadioMenuItem**,
264

operation, 262–266

`gtk_menu_item_new_with_label()`, 159,
264

`gtk_menu_item_new_with_mnemonic()`, 260,
262–263

`gtk_menu_item_remove_submenu()`, 265

`gtk_menu_item_set_submenu()`, 260, 265

GtkMenuShell

description, 262

example, 257–262

operation, 264–265

`gtk_menu_shell_append()`, 159, 260, 264

`gtk_menu_shell_insert()`, 265

`gtk_menu_shell_prepend()`, 265

GtkMessageDialog, 243

GtkNotebook

description, 139

example, 139–143

operation, 148–150

`gtk_notebook_append_page()`, 149

`gtk_notebook_append_page_menu()`,
142–143, 149

`gtk_notebook_insert_page()`, 149

`gtk_notebook_insert_page_menu()`, 149

`gtk_notebook_prepend_page()`, 149

`gtk_notebook_prepend_page_menu()`, 149

`gtk_notebook_remove_page()`, 149

GtkOptionMenu

description, 156

example, 157–161

operation, 162

`gtk_option_menu_get_history()`, 162

`gtk_option_menu_set_history()`, 159, 162

GtkPaned

- description, 139
- example, 139–143
- operation, 148

`gtk_paned_add1()`, 141, 148

`gtk_paned_add2()`, 142, 148

`gtk_paned_pack1()`, 148

`gtk_paned_pack2()`, 148

GtkProgressBar

- example, 122–126
- operation, 138

`gtk_progress_bar_pulse()`, 125–126, 138

GtkRadioButton

- description, 151
- example, 151–155
- properties and signals, 155–156

GtkRadioMenuItem. *See*

GtkMenuItem

`gtk_radio_menu_item_get_group()`, 264

`gtk_radio_menu_item_new_with_label()`,
264

`gtk_radio_menu_item_new_with_
mnemonic()`, 263–264

GtkRange, 166, 183. *See also* **GtkScale**;

GtkScrollbar

GtkRuler, 243

GtkScale

- description, 156
- example, 132–137, 157–161
- operation, 166–167

GtkScrollbar, 183. *See also*

GtkScrolledWindow

GtkScrolledWindow

- description, 183–184
- example, 184–185
- operation, 186

`gtk_scrolled_window_add_with_
viewport()`, 184–185

GtkSeparator

- description, 174
- example, 175–177
- operation, 178

GtkSeparatorMenuItem. *See*

GtkMenuItem

`gtk_separator_menu_item_new()`, 264

`GtkSignalFunc`, 267

GtkSizeGroup, 243

GtkSpinButton

- creating with Glade, 309
- description, 157
- example, 157–161
- operation, 167

GtkStatusbar

- example, 257–262
- operation, 268–269

`gtk_statusbar_pop()`, 268

`gtk_statusbar_push()`, 268

`gtk_statusbar_remove()`, 268

`gtk_statusbar_set_has_resize_grip()`,
269

GtkTable

- creating with Glade, 308
- description, 139
- example, 139–143
- operation, 146–148

`gtk_table_attach()`, 147

`gtk_table_attach_defaults()`, 142

GtkTextBuffer

- accessing text, 227
- child anchors, 238
- deleting text, 226
- description, 217
- emitting signals, 227
- example, 217–224
- extracting text information,
228–229
- indices, 224
- inserting text, 225–226
- modifying text, 226
- offsets, 224
- requirements, 224–225
- searching text, 231–232
- signals, 239–240
- text styles (*see* **GtkTextTag**)
- user actions, 226–227

`gtk_text_buffer_apply_tag()`, 236

`gtk_text_buffer_apply_tag_by_name()`,
221, 236

`gtk_text_buffer_begin_user_action()`,
226

`gtk_text_buffer_copy_clipboard()`, 239

`gtk_text_buffer_create_child_anchor()`,
222, 238

`gtk_text_buffer_create_mark()`, 232–233

`gtk_text_buffer_create_tag()`, 222, 234

- gtk_text_buffer_cut_clipboard(), 239
- gtk_text_buffer_delete(), 226
- gtk_text_buffer_delete_interactive(), 226
- gtk_text_buffer_delete_mark(), 233
- gtk_text_buffer_delete_mark_by_name(), 233
- gtk_text_buffer_delete_selection(), 226
- gtk_text_buffer_end_user_action(), 227
- gtk_text_buffer_get_char_count(), 227
- gtk_text_buffer_get_end_iter(), 222–223
- gtk_text_buffer_get_insert(), 233
- gtk_text_buffer_get_iter_at_child_anchor(), 238
- gtk_text_buffer_get_iter_at_mark(), 233
- gtk_text_buffer_get_line_count(), 227
- gtk_text_buffer_get_mark(), 233
- gtk_text_buffer_get_modified(), 226
- gtk_text_buffer_get_selection_bound(), 233
- gtk_text_buffer_get_selection_bounds(), 218
- gtk_text_buffer_get_slice(), 227
- gtk_text_buffer_get_tag_table(), 237
- gtk_text_buffer_get_text(), 227
- gtk_text_buffer_insert(), 225
- gtk_text_buffer_insert_at_cursor(), 226
- gtk_text_buffer_insert_interactive(), 225
- gtk_text_buffer_insert_interactive_at_cursor(), 226
- gtk_text_buffer_insert_pixbuf(), 223, 238
- gtk_text_buffer_insert_range(), 226
- gtk_text_buffer_insert_range_interactive(), 226
- gtk_text_buffer_insert_with_tags(), 236
- gtk_text_buffer_insert_with_tags_by_name(), 236
- gtk_text_buffer_iter_at_line(), 227
- gtk_text_buffer_iter_at_line_index(), 228
- gtk_text_buffer_iter_at_line_offset(), 228
- gtk_text_buffer_iter_at_offset(), 228
- gtk_text_buffer_iter_get_bounds(), 228
- gtk_text_buffer_iter_get_end_iter(), 228
- gtk_text_buffer_iter_get_selection_bounds(), 228
- gtk_text_buffer_iter_get_start_iter(), 228
- gtk_text_buffer_move_mark(), 233
- gtk_text_buffer_move_mark_by_name(), 233
- gtk_text_buffer_new(), 221, 225
- gtk_text_buffer_paste_clipboard(), 239
- gtk_text_buffer_place_cursor(), 232
- gtk_text_buffer_remove_all_tags(), 236
- gtk_text_buffer_remove_tag(), 236
- gtk_text_buffer_remove_tag_by_name(), 221, 236
- gtk_text_buffer_set_modified(), 226
- gtk_text_buffer_set_text(), 225
- GtkTextCharPredicate, 231
- GtkTextChildAnchor.**
 - See **GtkTextBuffer**, child anchors
- GtkTextIter
 - acquiring, 227–228
 - description, 225
 - example, 217–224
 - extracting text information, 228–229
 - scrolling to, 241
 - searching text, 231–232
 - setting position, 229–231
 - used with GtkTextTag, 237
- gtk_text_iter_backward_char(), 230
- gtk_text_iter_backward_chars(), 230
- gtk_text_iter_backward_cursor_position(), 230
- gtk_text_iter_backward_cursor_positions(), 230
- gtk_text_iter_backward_find_char(), 231
- gtk_text_iter_backward_line(), 230
- gtk_text_iter_backward_lines(), 230
- gtk_text_iter_backward_search(), 232
- gtk_text_iter_backward_sentence_start(), 231
- gtk_text_iter_backward_sentence_starts(), 231
- gtk_text_iter_backward_to_tag_toggle(), 231

- gtk_text_iter_backward_word_start(), 231
- gtk_text_iter_backward_word_starts(), 231
- gtk_text_iter_begins_tag(), 237
- gtk_text_iter_can_insert(), 229
- gtk_text_iter_ends_line(), 229
- gtk_text_iter_ends_sentence(), 229
- gtk_text_iter_ends_tag(), 237
- gtk_text_iter_ends_word(), 229
- gtk_text_iter_equal(), 218–220
- gtk_text_iter_forward_char(), 230
- gtk_text_iter_forward_chars(), 230
- gtk_text_iter_forward_cursor_position(), 219, 230
- gtk_text_iter_forward_cursor_positions(), 230
- gtk_text_iter_forward_find_char(), 231
- gtk_text_iter_forward_line(), 230
- gtk_text_iter_forward_lines(), 230
- gtk_text_iter_forward_search(), 232
- gtk_text_iter_forward_sentence_end(), 231
- gtk_text_iter_forward_sentence_ends(), 231
- gtk_text_iter_forward_to_end(), 231
- gtk_text_iter_forward_to_line_end(), 231
- gtk_text_iter_forward_to_tag_toggle(), 231
- gtk_text_iter_forward_word_end(), 231
- gtk_text_iter_forward_word_ends(), 231
- gtk_text_iter_get_buffer(), 228
- gtk_text_iter_get_bytes_in_line(), 229
- gtk_text_iter_get_char(), 227
- gtk_text_iter_get_chars_in_line(), 229
- gtk_text_iter_get_child_anchor(), 238
- gtk_text_iter_get_line(), 228
- gtk_text_iter_get_line_index(), 228
- gtk_text_iter_get_line_offset(), 228
- gtk_text_iter_get_offset(), 228
- gtk_text_iter_get_pixbuf(), 238
- gtk_text_iter_get_tags(), 237
- gtk_text_iter_get_toggled_tags(), 237
- gtk_text_iter_get_visible_line_index(), 228
- gtk_text_iter_get_visible_line_offset(), 228
- gtk_text_iter_has_tag(), 219–220, 237
- gtk_text_iter_inside_sentence(), 229
- gtk_text_iter_inside_word(), 229
- gtk_text_iter_is_cursor_position(), 229
- gtk_text_iter_is_end(), 229
- gtk_text_iter_is_start(), 229
- gtk_text_iter_set_line(), 229
- gtk_text_iter_set_line_index(), 230
- gtk_text_iter_set_line_offset(), 230
- gtk_text_iter_set_offset(), 229
- gtk_text_iter_set_visible_line_index(), 230
- gtk_text_iter_set_visible_line_offset(), 230
- gtk_text_iter_starts_line(), 229
- gtk_text_iter_starts_sentence(), 229
- gtk_text_iter_starts_word(), 229
- gtk_text_iter_toggles_tag(), 237

GtkTextMark

- cursor and, 232
- description, 225
- manipulating, 232–233
- scrolling to, 241
- selection, 233
- gtk_text_mark_get_buffer(), 234
- gtk_text_mark_get_deleted(), 233
- gtk_text_mark_get_left_gravity(), 234
- gtk_text_mark_get_name(), 234
- gtk_text_mark_get_visible(), 234
- gtk_text_mark_set_visible(), 233

GtkTextTag

- description, 234
- operation, 234–237
- used with GtkTextIter, 237
- gtk_text_tag_get_priority(), 236
- gtk_text_tag_set_priority(), 237

GtkTextTagTable, 225, 237–238

- gtk_text_tag_table_add(), 237
- GtkTextTagTableForeach, 238
- gtk_text_tag_table_foreach(), 238
- gtk_text_tag_table_get_size(), 238
- gtk_text_tag_table_lookup(), 237
- gtk_text_tag_table_new(), 221, 237
- gtk_text_tag_table_remove(), 237

GtkTextView

- creating, 240
- description, 217
- example, 217–224

GtkTextView, *continued*

- methods, 241–242
- properties, 240–241
- relationship with **GtkTextBuffer**, 240
- `gtk_text_view_add_child_at_anchor()`, 222, 238
- `gtk_text_view_backward_display_line()`, 242
- `gtk_text_view_backward_display_line_start()`, 242
- `gtk_text_view_backward_starts_display_line()`, 242
- `gtk_text_view_forward_display_line()`, 242
- `gtk_text_view_forward_display_line_end()`, 242
- `gtk_text_view_get_editable()`, 225
- `gtk_text_view_move_mark_onscreen()`, 241
- `gtk_text_view_move_visually()`, 242
- `gtk_text_view_new_with_buffer()`, 222, 240
- `gtk_text_view_place_cursor_onscreen()`, 241
- `gtk_text_view_scroll_mark_onscreen()`, 241
- `gtk_text_view_scroll_to_iter()`, 241
- `gtk_text_view_scroll_to_mark()`, 241
- `gtk_text_view_set_buffer()`, 240

GtkToggleButton, 243. *See also*

GtkCheckButton

GtkToolbar

- example, 257–262
- future of, 434
- operation, 266–268
- `gtk_toolbar_append_item()`, 267
- `gtk_toolbar_append_space()`, 267
- `gtk_toolbar_append_widget()`, 267
- `gtk_toolbar_insert_item()`, 267
- `gtk_toolbar_insert_space()`, 268
- `gtk_toolbar_insert_stock()`, 261, 267
- `gtk_toolbar_insert_widget()`, 267
- `gtk_toolbar_prepend_item()`, 267
- `gtk_toolbar_prepend_space()`, 267
- `gtk_toolbar_prepend_widget()`, 267
- `gtk_toolbar_remove_space()`, 268
- `gtk_toolbar_unset_style()`, 266

GtkTooltips

- example, 179–181
- operation, 178–179, 181
- `gtk_tooltips_disable()`, 181
- `gtk_tooltips_enable()`, 181
- `gtk_tooltips_set_tip()`, 178–180
- `GtkTreeCellDataFunc`, 205

GtkTreeIter

- description, 195
- example, 210–217
- operation, 199–200

GtkTreeModel (interface)

- description, 191–192
- methods, 196–200
- `gtk_tree_model_foreach()`, 198
- `GtkTreeModelForeachFunc`, 198
- `gtk_tree_model_get()`, 196, 213
- `gtk_tree_model_get_iter()`, 199, 212, 214
- `gtk_tree_model_get_iter_first()`, 200
- `gtk_tree_model_get_iter_from_string()`, 199
- `gtk_tree_model_get_path()`, 198
- `gtk_tree_model_get_value()`, 196
- `gtk_tree_model_iter_children()`, 200
- `gtk_tree_model_iter_has_child()`, 200
- `gtk_tree_model_iter_n_children()`, 200, 212
- `gtk_tree_model_iter_next()`, 200
- `gtk_tree_model_iter_parent()`, 200
- `gtk_tree_model_nth_child()`, 200
- `GtkTreePath`, 198. *See also*

GtkTreeStore, navigating

- `gtk_tree_path_append_index()`, 198
- `gtk_tree_path_compare()`, 199
- `gtk_tree_path_copy()`, 198, 212
- `gtk_tree_path_down()`, 199
- `gtk_tree_path_free()`, 198, 213–214
- `gtk_tree_path_get_depth()`, 199, 212
- `gtk_tree_path_get_indices()`, 199, 212
- `gtk_tree_path_is_ancestor()`, 199
- `gtk_tree_path_is_descendant()`, 199
- `gtk_tree_path_new()`, 198
- `gtk_tree_path_new_first()`, 198
- `gtk_tree_path_new_from_string()`, 198, 214
- `gtk_tree_path_next()`, 199
- `gtk_tree_path_prepend_index()`, 199

- gtk_tree_path_prev(), 199
- gtk_tree_path_to_string(), 198
- gtk_tree_path_up(), 199, 212
- GtkTreeRowReference**, 200
- gtk_tree_row_reference_free(), 200
- gtk_tree_row_reference_get_path(), 200
- gtk_tree_row_reference_new(), 200
- gtk_tree_row_reference_valid(), 200
- GtkTreeSelection**, 207–210
- GtkTreeSelectionForeachFunc, 209
- GtkTreeSelectionFunc, 210
- gtk_tree_selection_get_selected(), 208
- gtk_tree_selection_get_selected_rows(), 209
- gtk_tree_selection_get_view(), 208
- gtk_tree_selection_iter_is_selected(), 209
- gtk_tree_selection_path_is_selected(), 209
- gtk_tree_selection_select_all(), 209
- gtk_tree_selection_selected_foreach(), 208–209
- gtk_tree_selection_select_iter(), 209
- gtk_tree_selection_select_path(), 209
- gtk_tree_selection_select_range(), 209
- gtk_tree_selection_set_mode(), 208
- gtk_tree_selection_set_select_function(), 210
- gtk_tree_selection_unselect_all(), 209
- gtk_tree_selection_unselect_iter(), 209
- gtk_tree_selection_unselect_path(), 209
- GtkTreeStore**
 - accessing nodes, 196–198
 - creating, 196
 - description, 192
 - example, 210–217
 - extracting information, 197–198
 - navigating, 198–200
 - removing nodes, 197
 - setting and adding nodes, 196–197
 - traversing, 198
- gtk_tree_store_append(), 197, 212–213
- gtk_tree_store_clear(), 197
- gtk_tree_store_insert(), 196
- gtk_tree_store_insert_after(), 197
- gtk_tree_store_insert_before(), 197
- gtk_tree_store_is_ancestor(), 197
- gtk_tree_store_iter_depth(), 198
- gtk_tree_store_new(), 195–196, 214
- gtk_tree_store_prepend(), 197
- gtk_tree_store_remove(), 197
- gtk_tree_store_set(), 196, 214
- gtk_tree_store_set_value(), 196
- GtkTreeView**
 - description, 192
 - example, 192–195, 210–217
 - future of, 435
 - manipulating, 201
 - properties, 202
 - selection (*see* **GtkTreeSelection**)
 - signals, 202
- gtk_tree_view_append_column(), 194, 203, 216
- gtk_tree_view_collapse_all(), 201
- gtk_tree_view_collapse_row(), 201
- GtkTreeViewColumn**
 - cell renderers and, 202–206
 - description, 192
 - example, 192–195
 - manipulating, 203–206
 - properties and signals, 205–206
- gtk_tree_view_column_add_attribute(), 203
- gtk_tree_view_column_clear(), 203
- gtk_tree_view_column_clear_attributes(), 203
- gtk_tree_view_column_new_with_attributes(), 194–195, 203, 215
- gtk_tree_view_column_pack_end(), 203
- gtk_tree_view_column_pack_start(), 203
- gtk_tree_view_columns_autosize(), 204
- gtk_tree_view_column_set_attributes(), 203
- gtk_tree_view_column_set_cell_data_func(), 205, 215
- gtk_tree_view_expand_all(), 201
- gtk_tree_view_expand_row(), 201
- gtk_tree_view_get_column(), 204
- gtk_tree_view_get_columns(), 204
- gtk_tree_view_get_cursor(), 201
- gtk_tree_view_get_selection(), 208
- gtk_tree_view_insert_column(), 204
- gtk_tree_view_insert_column_with_attributes(), 204

- gtk_tree_view_insert_column_with_data_func(), 204–205
- gtk_tree_view_move_column_after(), 204
- gtk_tree_view_remove_column(), 204
- gtk_tree_view_row_activated(), 201
- gtk_tree_view_row_expanded(), 201
- gtk_tree_view_scroll_to_cell(), 201
- gtk_tree_view_set_cursor(), 201
- GtkVBox**. *See* button boxes; **GtkBox**
- GtkViewport**, 184
- GtkVPaned**. *See* **GtkPaned**
- GtkVScale**. *See* **GtkScale**
- GtkVScrollbar**. *See* **GtkScrollbar**
- GtkVSeparator**. *See* **GtkSeparator**
- GtkWidget**
 - description, 118
 - destroying, 119–120, 144
 - disabling, 119
 - displaying, 118–119
 - hiding, 119–120
 - methods, 118–119
 - properties, 119–120
 - signals, 120
- gtk_widget_add_accelerator(), 182
- gtk_widget_destroy(), 116–119
- gtk_widget_hide(), 119
- gtk_widget_hide_all(), 119
- gtk_widget_hide_on_delete(), 117
- gtk_widget_show(), 118
- gtk_widget_show_all(), 116–117, 119, 266
- gtk_widget_show_now(), 119
- GtkWindow**
 - creating, 117
 - destroying, 122
 - icons, 122
 - positioning, 121
 - properties, 121–122
 - resizing, 121–122
 - titles, 120–121
- gtk_window_add_accel_group(), 182, 259
- gtk_window_remove_accel_group(), 182
- gtk_window_set_default_icon_from_file(), 122
- gtk_window_set_icon(), 122
- gtk_window_set_icon_from_file(), 122
- gtk_window_set_icon_list(), 122
- GTraverseFunc, 51
- GTree
 - adding and replacing nodes, 48–49
 - creating, 46–48
 - deleting nodes, 50
 - description, 46
 - example, 47–51
 - finding nodes, 49–50
 - gathering statistics, 51–52
 - traversing, 51
- g_tree_destroy(), 52
- g_tree_foreach(), 51
- g_tree_height(), 51
- g_tree_insert(), 48–49
- g_tree_lookup(), 49–50
- g_tree_lookup_extended(), 49–50
- g_tree_new(), 47
- g_tree_new_full(), 47–48
- g_tree_new_with_data(), 47
- g_tree_nnodes(), 51
- g_tree_remove(), 50
- g_tree_replace(), 48
- g_tree_steal(), 50
- g_try_malloc(), 8
- g_try_realloc(), 8
- g_type_add_interface_static(), 106–107
- G_TYPE_CHECK_CLASS_CAST(), 65–66
- G_TYPE_CHECK_CLASS_TYPE(), 65–66
- G_TYPE_CHECK_INSTANCE_CAST(), 65–66
- G_TYPE_CHECK_INSTANCE_TYPE(), 65–66
- G_TYPE_FROM_INTERFACE(), 108
- GTypeInfo, 67–69, 105
- G_TYPE_INSTANCE_GET_CLASS(), 65–66
- GTypeInterface**. *See* interfaces, creating
- G_TYPE_OBJECT. *See* type identifier, base class
- g_type_register_static(), 67–69
- g_ucs4_to_utf8(), 22
- g_ucs4_to_utf16(), 21
- g_unichar_digit_value(), 18
- g_unichar_isalnum(), 16
- g_unichar_isalpha(), 16
- g_unichar_iscntrl(), 17
- g_unichar_isdefined(), 16
- g_unichar_isdigit(), 16
- g_unichar_isgraph(), 17
- g_unichar_islower(), 16

- `g_unichar_isprint()`, 17
- `g_unichar_ispunct()`, 16
- `g_unichar_isspace()`, 17
- `g_unichar_istitle()`, 16
- `g_unichar_isupper()`, 16
- `g_unichar_iswide()`, 17
- `g_unichar_isxdigit()`, 16
- `g_unichar_tolower()`, 18
- `g_unichar_totitle()`, 18
- `g_unichar_toupper()`, 18
- `g_unichar_to_utf8()`, 22
- `g_unichar_type()`, 17
- `g_unichar_validate()`, 16
- `g_unichar_xdigit_value()`, 18
- `g_utf8_casefold()`, 20
- `g_utf8_collate()`, 20
- `g_utf8_collate_key()`, 21
- `g_utf8_find_next_char()`, 19
- `g_utf8_find_prev_char()`, 19
- `g_utf8_get_char()`, 19
- `g_utf8_get_char_validated()`, 18
- `g_utf8_next_char()`, 19
- `g_utf8_normalize()`, 20
- `g_utf8_offset_to_pointer()`, 19
- `g_utf8_pointer_to_offset()`, 19
- `g_utf8_prev_char()`, 19
- `g_utf8_strchr()`, 20
- `g_utf8_strdown()`, 20
- `g_utf8_strlen()`, 19
- `g_utf8_strncpy()`, 19
- `g_utf8_strrchr()`, 20
- `g_utf8_strup()`, 20
- `g_utf8_to_ucs4()`, 21
- `g_utf8_to_ucs4_fast()`, 21
- `g_utf8_to_utf16()`, 21
- `g_utf8_validate()`, 18
- `g_utf16_to_ucs4()`, 21
- `g_utf16_to_utf8()`, 21
- GValue, 75–77
- GValueArray, 109
- `g_value_dup_string()`, 76
- `g_value_get_*`(***), 76–77
- `G_VALUE_HOLDS_*`(***), 76–77
- `g_value_set_*`(***), 76–77
- `g_value_set_static_string()`, 76
- `g_value_unset()`, 77
- `g_vsnprintf()`, 14
- `g_warning()`, 24–25

H

- hash table. *See* `GHashTable`
- header file
 - `GConf`, 350
 - `Glade`, 315
 - `GLib`, 6
 - `GNOME`, 247
 - `GnomeVFS`, 379
 - `GTK+`, 114
- help
 - displaying GNOME documents, 254
 - generating documentation, 337–338
 - widget tooltips (*see* `GtkTooltips`)
- high scores. *See* `GnomeScores`; scores
- hooks. *See* signals, hooks
- hyperlinks. *See* `GnomeHRef`; URL, displaying

I

- icons
 - choosing (*see* `GnomeIconEntry`)
 - `GtkWindow` (*see* `GtkWindow`, icons)
 - themes, 435
- images. *See* `GdkPixbuf`; `GnomePixmapEntry`; `GtkImage`; `GtkTextBuffer`, child anchors
- include file. *See* header file
- indexing arrays and lists, 4
- inheritance
 - defining subclasses, 99–104
 - description, 61–62
 - multiple, 63
- input focus, 119–120, 309
- instance
 - definition, 60
 - structure, 64–65, 68–69, 79, 99
- interfaces
 - base initializer, 106
 - creating, 104–109
 - description, 63–64
 - implementing and installing, 106–108
 - using, 108–109

intltool, 327, 329–330, 333, 341
INTLTOOL_DESKTOP_RULE, 331–332
IRC, 434
iterators
 text (*see* `GtkTextIter`)
 tree (*see* `GtkTreeIter`)

K

keyboard operation, 181–182. *See also*
 accelerators; mnemonics
keys
 GConf (*see* `GConf`, keys)
 hash, 36 (*see also* `GHashTable`)
 sorting, 20–21
Kimball, Spencer, 111

L

label, 122. *See also* **GtkLabel**
libbonobo, 246
libbonobo-activation, 246
libbonoboui, 246
libegg, 434
libgconf-2, 344
libglade, 312–320
libgnome, 245, 247
libgnomecanvas, 245
libgnomeui, 245, 257
libgnomevfs, 245
libtool, 333
limits of this book, 2
line splitting, 3
list box. *See* tree and list widgets
lists. *See* `GList`
locale, 20, 22–23, 128. *See also* `gettext`;
 intltool
log domain, 24

M

m4, 328, 334
mailing lists, 433–434
main event loop, 114
Makefile templates. *See* *Makefile.am*;
 Makefile.in
Makefile.am, 327, 331–333
Makefile.in, 327
marks. *See* **GtkTextMark**

markup language. *See* Pango, markup
 language
marshallers. *See* signals, marshallers
Mattis, Peter, 111
membership relations, 61–63
memory chunks
 creating and deallocating, 9–11
 diagnostics, 11
 example, 9–11
 used in a tree, 47–50
memory management, 7–12
menu bars, 257. *See also* **GtkMenuBar**
menus
 GTK+ (*see* **GtkMenu**;
 GtkMenuBar;
 GtkMenuItem;
 GtkMenuShell)
 option (*see* **GtkOptionMenu**)
 popup (*see* context menus)
message logging, 24–27, 57
methods
 access (*see* access methods)
 creating, 72
 definition, 60
MIME types
 declaring, 414–416
 determining, 413–414
mnemonics, 128, 160, 181–182,
 262–263
modal windows, 121
models. *See* MVC
multiple inheritance, 63
MVC, 157, 183, 191, 201, 217

N

naming conventions, 6, 60
news, 430
notebooks. *See* **GtkNotebook**
numeric data entry. *See*
 GtkSpinButton

O

object-oriented programming, 59–64
objects
 creating, 81–82
 data pointer, 133–136
 definition, 60

objects, *continued*
using, 81–84
OMF files. *See* help, generating
documentation
option menus. *See* **GtkOptionMenu**

P

packing (widgets), 112–113, 143–149,
306–307
paned widgets. *See* **GtkPaned**
Pango
description, 112
GtkTextTag and, 234–235
markup language, 127–128
parameters
command-line, 248 (*see also*
session management)
GObject (*see* GParamSpec)
paths. *See* **GtkTreeStore**, navigating
PKG_CHECK_MODULES, 329–330
pkg-config
compiler and linker options,
325–326
package status, 325
purpose, 323–324
with Makefile, 326
PKG_CONFIG_PATH, 324–325
platform for this book, 3
pointers
required knowledge, 1, 4
untyped, 6–7
popup menus. *See* context menus
preferences. *See* GConf
prerequisites for this book, 1–2
print handlers, 26–27
progress bar, 122. *See also*
GtkProgressBar
properties
and the base class, 71, 78
child (*see* child properties)
declaring in classes, 73–81,
102–104
definition, 72–73
in Glade (*see* Glade, property
editor)
installing in classes, 77–80
internal access, 78–80, 102–103

justification, 80–81
using, 82–83, 103–104

Q

quarks, 11–12

R

radio buttons. *See* **GtkMenuItem**;
GtkRadioButton
reference counts, 60–61
references
example, 83–84
floating, 118, 143
strong, 83–84
weak, 83–84
renderer. *See* **GtkCellRenderer**
resizing. *See* **GtkWindow**, resizing

S

saving program state. *See* session
management
scores, 256. *See also* **GnomeScores**
scrolling, 183. *See also*
GtkScrolledWindow;
GtkTextView; **GtkTreeView**
ScrollKeeper, 338–339
searching
GtkTextBuffer, 231–232
strings, 14–15
Unicode strings, 19–20
selection
text buffer (*see* **GtkTextMark**,
selection)
trees (*see* **GtkTreeSelection**)
session management, 246, 298.
See also **GnomeClient**
shell, using to execute programs,
252–253
signal handlers
attaching, 93–95, 99
defining, 85–88
description, 84–85
example, 87–88, 93–94
signals
aborting emission, 98–99
accumulators, 85, 87, 92–93
blocking, 98–99

- signals, *continued*
 - cache array, 85, 89
 - dash and underscore in names, 117
 - defining, 85–88, 99
 - description, 84–85
 - details, 95
 - emitting, 84, 88–89
 - events and, 114, 117–118
 - extracting identifiers, 98–99
 - Glade and (*see* Glade, signals)
 - handler (*see* signal handlers)
 - hooks, 89, 96–97
 - listing and printing, 98–99
 - marshallers, 85, 87, 89–92
 - property change (notification), 72
 - using (*see* signal handlers, attaching; signals, emitting)
- signature, 88
- sliders. *See* **GtkScale**
- sorting
 - GArray, 46
 - GList, 42–43
 - quicksort, 57
- sound, 254–256, 436
- spin button. *See* **GtkSpinButton**
- standard event handlers, 115, 117
- status bars, 257. *See also* **GtkStatusBar**
- stock items, 130, 137–138, 286
- strings
 - array operations, 15
 - converting to and from numbers, 15
 - copying, 13–14
 - creating, 13–14
 - GString (*see* GString)
 - quarks (*see* quarks)
 - searching, 14–15
 - splitting and joining, 13, 15
 - translating, 14–15
 - Unicode (*see* Unicode)
- strnfill(), 13
- subclasses. *See* inheritance
- SUBDIRS, 331–332
- synchronous file access
 - description, 380–381
 - example, 381–383

- file information, 385–388
- file management, 388–389
- opening and closing files, 383–384
- reading, writing, seeking, 384–385
- relationship to Unix file operations, 380–381

T

- table containers. *See* **GtkTable**
- tabs. *See* **GtkNotebook**
- tag tables. *See* **GtkTextTagTable**
- tags. *See* **GtkTextTag**
- text styles. *See* **GtkTextTag**; Pango, markup language
- text widgets. *See* **GnomeEntry**; **GtkCombo**; **GtkEntry**; **GtkTextBuffer**; **GtkTextView**
- threads, 58
- timeouts, 224
- timer. *See* GTimer
- titlecase, 16–18
- titles. *See* Glade, window titles; **GnomeApp**, titles; **GtkWindow**, titles
- toolbars, 257. *See also* **GtkToolbar**
- toolkits, 111
- tooltips. *See* **GtkTooltips**
- transfers. *See* GnomeVFS, transfers
- tree and list widgets, 191. *See also* **GtkTreeModel**
- trees. *See* **GtkTreeModel**; GTree
- type identifier
 - base class, 70
 - creating objects with, 81–82
 - function, 67–69, 100–101
 - in type identifier function, 67–68
 - in utility macro, 65–66
 - parent class, 100
- typographical conventions, 2–3

U

- Unicode
 - character encodings, 16

Unicode, *continued*

- converting strings and characters, 18–23
- encoding conversion, 21–23
- error codes, 22–23
- extracting string information, 19
- GString and, 34
- GTK+ and, 118
- identifying, 16–18
- in **GtkTextBuffer**, 225
- operations, 15–23
- searching and navigating strings, 19–20
- UCS-4, 16
- UTF-8, 16
- UTF-16, 16
- validating, 16, 18–19

URI. *See* GnomeVFS, URI

URL, displaying, 253. *See also* **GnomeHRef**

user data pointer

- iteration and, 41–43
- signal handler example, 152–154
- signal handlers and, 94

user interface guidelines

- combo boxes, 165
- dialogs, 191
- druids, 292
- entry boxes, 276
- frames, 178
- GtkLabel**, 129
- menu bars, 262
- modal windows, 121
- option menus, 156
- origin, 112
- preferences, 375
- sliders, 167
- sound, 254
- status bars, 269
- toolbars, 266
- tree view selections, 210
- widget arrangement, 151
- window titles, 120–121

user interfaces, graphically

- creating, 303. *See also* Glade

V

value containers. *See* GValue

views. *See* MVC

W

watermarks, 293, 295

widgets

- containers, 138–139 (*see also* button boxes; **GtkAlignment**; **GtkBox**; **GtkPaned**; **GtkTable**)

- data entry, 156–174 (*see also*

- GnomeColorPicker**;

- GnomeFileEntry**;

- GnomeFontPicker**;

- GtkColorSelection**;

- GtkCombo**; **GtkEntry**;

- GtkFileSelection**;

- GtkFontSelection**;

- GtkOptionMenu**; **GtkScale**;

- GtkSpinButton**;)

- date and time (*see*

- GnomeDateEdit**)

- default, 120

- definition, 112–113

- display, 122 (*see also* **GtkImage**;

- GtkLabel**; **GtkProgressBar**)

- drawing, 113

- in text buffers

- (*see* **GtkTextBuffer**,

- child anchors)

- operations (*see* **GtkWidget**)

- tree, 112, 311

- windows (*see* **GtkWindow**)

windows. *See* **GnomeApp**; **GtkWindow**

wizards. *See* druids