

# Index

## Symbols and Numerics

- \ (backslash) directory separator, 584
- | (bar) OR operator, 804, 846
- : (colon)
  - exchange manager URL separator, 637
  - table item, suppressing display in, 392, 421
- , (comma) Bluetooth address separator, 639
- 8-bit color, 319–320
- 5-way navigation button programming, palmOne, 813–816, 859
- ? (question mark) exchange manager URL prefix, 637
- ;(semicolon) exchange manager URL separator, 637
- 16-bit
  - Garnet memory reference, 827
  - port number, 713
- 68K application
  - emulating 68K environment, 12–13
  - exchanging data, 620
  - Motorola processor, 859
  - Palm OS Protein, porting to, 14
- 64K code resource limit, 827, 830–834
- / (slash)
  - directory separator, 584
  - root directory indicator, 569
- // (slash, double) exchange library URL prefix, 639
- 32-bit IP address, 709, 711
- 32K jump limit, 827, 828–830
- \_\_ (underscore, double) system function prefix, 832

## A

- About Librarian dialog box (Librarian), 194, 195
- AboutForm resource, 95
- acceptedSocket field, 620

- access mode
  - database, 476–477, 481, 490
  - directory, 594, 595
  - file, 594, 595, 610, 852
- AddCRs method, 780
- AddRec method, 773
- Address Book conduit, 792
- Advanced RISC Machine (ARM), 5, 12–13, 14, 712
- AFX\_MANAGE\_STATE macro, 757
- alarm programming, 359–366, 374. *See also* alert
- AlarmFormHandleEvent function, 365–366
- AlarmTriggered function, 364
- alert. *See also* alarm programming; attention manager
  - audio, 19
  - dialog box
    - button, 158, 166, 200–201
    - color, 336
    - confirmation, 19
    - creating, 157–158, 165–166
    - displaying, 199–200
    - error, 19
    - file data, displaying in, 603
    - Help ID value, 158, 165
    - information-only, 18
    - message text, 158, 166, 201–202
    - resource database reference, 200, 201
    - text field, displaying using, 95–97
    - title bar, 158, 166
    - type, assigning, 158, 165
    - warning, 19
  - VFS Test application, 585
- AlmGetAlarm function, 361
- AlmSetAlarm function, 360, 361, 364, 366
- alpha blending, 344

**anti-aliasing, 344–345**

**API (Application Program Interface), 859. See also specific API**

**app info string list, 859**

AppCallWithCommand **macro, 301**

appData **field, 619, 663**

AppendBlankRecord **method, 795**

AppendDuplicateRecord **method, 793–795**

AppEventLoop **function, 82, 83, 294–295**

AppHandleEvent **handler, 83, 84, 85–86**

appInfoID **field, 471**

AppLaunchCmd.h **file, 301**

AppLaunchWithCommand **macro, 301**

**application launcher, 298–299, 860**

**Application Program Interface (API), 859. See also specific API**

AppResources.xrd **file, 176, 178**

AppStart **routine, 81**

AppStop **routine, 82**

**arc, drawing, 347**

**ARM (Advanced RISC Machine), 5, 12–13, 14, 712**

**Arrange ⇄ Align Centers Horizontally (Constructor), 172**

**Arrange ⇄ Align Centers Vertically (Constructor), 172**

**Arrange ⇄ Arrange Objects (Constructor), 187**

**Arrange Objects dialog box (Constructor), 187**

**Arrange ⇄ Spread Horizontally in Container (Constructor), 172**

**Arrange ⇄ Spread Vertically in Container (Constructor), 172**

**array**

Sony CLIE JogAssist mask, 821

string array, passing to list, 243

**Assign method, 799**

**attention manager**

alarm, relation to, 374

audio, 377, 379

database record, displaying using, 379

dialog box, displaying using

detail dialog box, 374–375

interrupting current operation, 377

list dialog box, 375, 379, 384

modal dialog box, 374

disabling attention indicator, 384–385

draw request, 382–383, 384

frequency of attention attempt, 378

Go There request, 383

Got It request, 384

launch code, responding to, 375, 376, 378–381

LED, flashing, 377

list item, adding using, 377

removing item, 384

system preference, overriding, 377–378

vibrating handheld, 377

AttentionMgr.h **file, 377**

AttnCommandArgsDrawDetailTag **structure, 382**

AttnCommandArgsDrawListTag **structure, 383**

AttnCommandArgsType **structure, 380, 382, 383**

AttnForgetIt **function, 384**

AttnGetAttention **function, 376, 378, 380, 384**

AttnIndicatorEnable **function, 384**

AttnIndicatorEnabled **function, 384**

AttnLaunchCodeArgsType **structure, 379–380**

AttnListOpen **function, 384–385**

AttnUpdate **function, 384**

attributes **field, 471, 472, 507**

**audio**

alarm, 360, 364

alert, 19

attention manager, 377, 379

compression, 352

generating tone manually, 350–351

handle, 352, 353

MIDI, 351, 873, 881

POSE audio setup, 124

resource, adding to, 163

sampled sound, 352–353

Sound Manager, 349–353, 882

speaker, 349–350

stereophonic, 352

stopping, 351

synchronous/asynchronous, 351, 353

system sound, 350, 364

volume, 24, 352

**auto-off timer, 860. See also sleep mode**

## B

**Background Process, 16**

**backlight feature, 272, 293, 860**

**backslash (\) directory separator, 584**

**backup**

application preference, 494

conduit, 8–9, 481, 484, 730

database, 68, 481, 483, 729

HotSync configuration, 735

location, 569

**bar (|) OR operator, 804, 846**

**battery**

- changing battery setting, 304
- debugging, simulating low when, 118
- event, low battery, 306
- information about, returning, 304, 305
- keyDownEvent triggered in reaction to low, 306
- life expectancy, 5
- lithium ion, 11
- voltage, 304
- warning threshold, 304–305

**baud rate, 679, 680, 681, 695, 700****beaming data. See exchanging data****Berkeley sockets API**

- Cobalt support, 722, 723
- error handling, 706, 707
- Garnet support, 703, 722
- header file, 706
- network library
  - function, mapping to network library routine, 706
  - referencing, 707
- pipe, 705
- timeout, 707
- UNIX compatibility, 722

**bfields.h file, 798****big-endian byte order, 712, 860****bitmap**

- button, 182, 186, 249–250
- color, 334–335
- defined, 860
- drawing, 327, 333–334
- family, 334, 861
- font, 273–275, 282, 838
- form, 184, 868
- handle, 333
- icon, 25, 162–163, 166
- pointer, 333
- slider, 189

**blitter, 340, 861****Bluetooth technology**

- address format, 639
- baud rate, 679
- exchanging data, 638, 639–640
- frequency, 861
- introduced, 38
- range, 861
- serial connection, 679
- startup task, 680

**BmpGetColorTable function, 324****Book Details dialog box (Librarian), 192–193, 544****Bring Up to Date command (CodeWarrior), 61****BtExgLib.h file, 638****built-in application, 861****busy bit, 515, 516, 520, 537, 861****busy field, 472****button**

- alert dialog box button, 158, 166, 200–201
- anchoring, 182
- application button, overriding, 294–297
- bitmap, 182, 186, 249–250
- command toolbar button, adding, 249
- defined, 861
- ease of use, designing for, 31
- font, 218
- form, adding to dynamic, 841
- frame, 182, 868
- hardware button, 870
- ID, 181
- justification, 35
- label, 31, 182, 186, 188, 218
- placement, 35, 249
- push button
  - aligning, 187
  - bitmap, 186
  - copying, 187
  - defined, 877
  - group, 23, 186, 219–220
  - label, 186
  - sizing, 187
  - speed, 29
  - status, returning, 217, 219
  - status, setting, 187, 219
- repeating, 22, 188, 212, 878
- silkscreen button, 32, 881
- sizing, 37, 187
- spacing, 35
- visibility, 217

**byte order**

- ARM, 712
- big-endian, 712, 860
- defined, 861
- host, 712, 870
- Intel, 803
- IP address byte order, changing, 711–712
- little-endian, 712, 803, 872
- TCP/IP, 712

**C****CAddressRecord class, 792****callback function, 861**

**card**

expansion card

- application, launching upon inserting, 569–570
- application resident on, running, 570–571
- database, transferring to/from, 615
- emulating, 132–133
- event handling, 573
- file stored on, accessing, 354
- flash memory, 569, 573
- insertion, detecting, 572–573
- memory module card versus, 256
- notification, 573–576
- removal, detecting, 572–573
- volume, 568
- writing data to both RAM and expansion card, 599

memory module

- address space, 256, 259
- AlmSetAlarm function cardNo parameter, 360
- AttnGetAttention function cardNo parameter, 376
- databases on card, returning number of, 478
- Dm functions cardNo parameter, 473–474, 483
- expansion card versus, 256
- LocalID, 470–472, 478, 479, 537, 872
- logical, 255–256
- SysAppLaunch function cardNo parameter, 300
- SysGetROMToken function cardNo parameter, 306
- SysNotifyRegister function cardNo parameter, 574

**Carrier Detect (CD) signal, 691**

**Cascading Style Sheets (CSS), 280, 284**

**catalog resource, 861**

category **field, 472**

CategoryEdit **function, 545**

CategoryEditV10 **function, 545**

CategoryEditV20 **function, 545**

CategoryGetNext **function, 546, 547**

CategoryInitialize **function, 539**

CategorySelect **function, 543–544, 545**

CategorySelectV10 **function, 545**

CategorySetName **function, 548**

CategorySetTriggerLabel **function, 547**

CBaseConduitMonitor **class, 755**

CBaseDTLinkConverter **class, 789, 790, 803**

CBaseIterator **class, 790**

CBaseMonitor **class, 789, 790, 791–792**

CBaseRecord **class, 790, 793, 796**

CBaseSchema **class, 790, 796**

CBaseTable **class, 790, 792–793, 795**

CBoolField **class, 798**

**C/C++ Sync Suite development environment, 725, 730**

CCategoryMgr **class, 789, 790**

**CD (Carrier Detect) signal, 691**

CDateField **class, 798**

CDbCreateDB **class, 805–806**

**CDK (Conduit Development Kit)**

- base class support, 731
- Conduit Configuration tool, 732–735
- downloading, 725
- header files
  - bfields.h, 798
  - Condapi.h, 753, 761
  - CPalmRec.h, 782
  - eDbOpenModes constant, 804
  - pcmgr.h, 772
  - syncmgr.h, 753, 755, 760, 800
  - tables.h, 798

**CDMA (Code Division Multiple Access), 862**

CfgConduit **function, 750, 758, 759, 761, 762–763**

CfgConduitInfoType **structure, 761–762**

**Change HotSync Action dialog box (Conduit Wizard), 748**

**Change Security dialog box (Librarian), 194**

ChangeRecord **function, 522**

**character. See also text**

- digit, testing if character is, 286, 844
- encoding, 843, 845–846, 849–850, 862, 873
- macro, 285–288
- size in bytes, returning, 848
- text buffer
  - returning character from, 847–849
  - transliterating character in, 845

Chars.h **file, 817**

CharShiftJIS.h **file, 846**

**checkbox**

- defined, 862
- form, adding to dynamic, 841
- group, 24, 182, 219–220
- label, 23, 182
- mutually exclusive, 182
- status
  - returning, 217, 219
  - setting, 182, 206
- table, inserting in, 389, 390, 399

checkboxTableWidgetItem **style, 389, 390**

CHHMgr **class, 766, 767, 768**

CHotSyncActionDlg **object, 759**

ChrHorizEllipsis **macro, 279**

- chunk (memory unit)**
  - database, 258, 261
  - described, 256, 862
  - record, 258, 469, 516–517, 518, 523
- CIntegerField **class**, **798**
- circle, drawing**, **347**
- clear to send (CTS) signal**, **674, 695, 696, 697**
- CLibPcMgr **class**, **782**
- CLiÉ handheld. See Sony CLiÉ handheld**
- clipboard, 307–309**
  - ClipboardAddItem **function**, **308–309**
  - ClipboardAppendItem **function**, **309**
  - ClipboardGetItem **function**, **309**
- clipping region**, **347, 862**
- clock**, **367**
- Close **method**, **772**
- CloseDB **method**, **772**
- CloseSerial **function**, **682–683**
- close-wait state, network library**, **722, 862**
- CmConduitType **structure**, **736–737**
- CmGetCorePath **function**, **736**
- CmInstallConduit **function**, **736, 737**
- CMP (Connection Management Protocol)**, **675**
- CmRemoveConduitByCreatorID **function**, **738**
- CmSetCreator **functions**, **737**
- CMyCondnItArchIueMgr **class**, **767**
- CMyCondnItBackupMgr **class**, **767**
- CMyCondnItPcMgr **class**, **767, 768**
- CMyCondnItRecord **class**, **767**
- CMyCondnItSync **class**, **767**
- CMyConduitPcMgr **class**, **767, 779–781**
- CMyConduitRecord **class**
  - AppendDuplicateRecord method implementation, 793
  - Assign method, 799
  - coding, initial, 768–769
  - comparison operator, 799
  - ConvertGenericToPc method implementation, 779–780
  - converting CMyConduitRecord object to handheld format, 801–802
  - GetData method, 797
  - introduced, 766
  - IsPrivate method, 797
  - SetData method, 798
- cnvFrom68KApp **field**, **620**
- Cobalt Palm OS**
  - ARM support, 12–13
  - Berkeley sockets support, 722, 723
  - form, loading in, 81
  - graphics model, 343–347
  - header file, 111, 113
  - International Feature Set support, 286
  - kernel, 15–16
  - memory architecture, 259–261, 264
  - Movie Server, 353
  - PalmOSGlue library support, 13, 217, 219
  - processor, 12
  - time macros, 368–369
- code**
  - island, 830, 862
  - rearranging to beat 32K jump limit, 829
  - resource 64K limit, 827, 830–834
  - section, 862
  - Smart code model, 829
- Code Division Multiple Access (CDMA)**, **862**
- CodeWarrior IDE**
  - assembler, 46
  - Bring Up to Date command, 61
  - compiling application using, 46, 55, 61–62
  - constructor, 47
  - debugger, 46, 140–143
  - linking, 46, 56, 61–62, 219, 828–829
  - Make command, 61
  - New dialog box, 48–49, 50
  - New Project dialog box, 60, 830
  - New Target dialog box, 57, 833
  - out of range error, 828
  - Palm OS Application Wizard, 51–53, 830
  - PalmRez plug-in, 46, 59, 60, 875
  - PalmSim, 47
  - PiIRC plug-in, 47, 539, 836–839
  - project
    - creating, 48–53
    - default, 62
    - defined, 877
    - Files view, 54–56
    - naming, 49, 50, 51
    - opening, 48
    - resource, 47, 877
    - saving, 58
    - source code reference, 47
    - Starter project, 50
    - stationery, 49–51, 877
    - Targets view, 56–58
    - window, 53–58, 877
  - Rez tool, 878
  - segmenting application, 830–832

## CodeWarrior IDE (continued)

### target

- changing, 54
- creating, 57, 833
- debugging, 141
- default, 62
- defined, 883
- displaying, 55
- editing, 58–61
- multiple targets, working with, 47, 56–57
- PRC file setup, 60
- project Targets view, 56–58

### version

- full version, 725
- Mac OS version, 117

## colon (:)

- exchange manager URL separator, 637
- table item, suppressing display in, 392, 421

## color

- alert dialog box, 336
- alpha blending, 344
- background, 317, 326, 327–328, 335–336, 337
- bitmap, 334–335
- border, 335, 336, 337
- cursor, 336
- debug ROM image, 337
- depth, 319–323, 863
- direct, 320, 335
- 8-bit, 319–320
- foreground, 327–328, 335, 337
- form, 336
- gradient, 345
- grayscale, 161, 162, 319
- icon, 160–161, 162, 167, 326
- index value, 325–326
- inverting, 317, 326, 328, 329, 333
- line, 328–329
- menu, 335, 337
- monochrome, 319
- palette, 319, 324–325, 335
- picker, 225–226
- rectangle, 329
- rendering state, 345
- RGB, 323–324, 325–326, 327, 337, 339
- slider, 225–226
- table, 323–325, 335, 863
- text, 226, 327–328, 336
- transparency, 344
- user interface, 335–339
- window, 317, 318

## comma (,) Bluetooth address separator, 639

## command toolbar, 19–20, 248–253, 873

## comment, form, 179

## compilation

- CodeWarrior, using, 46, 55, 61–62
- conditional, 147
- Eclipse Workbench, using, 62, 74
- error handling, 147–148
- PalmOSGlue library, using, 850

## componentIndex field, 620

## compression

- audio, 352
- icon, 162

## Condapi.h file, 753, 761

## CondCfg.exe file, 732

## CondMgr.dll file, 735–736

## CondSwitch HotSync command, 735

## CondSwitch.exe file, 735

## conduit. See also synchronization

- action, default, 753
- Address Book, 792
- application interfaced with, specifying, 745–746
- archiving, 748, 764, 777
- backup conduit, 8–9, 481, 484, 730, 860
- category support, 748
- data transfer type, specifying, 746–747
- database, identifying to, 734
- defined, 863
- development environment, choosing, 725, 730–731
- dialog box display, 758, 759
- DLL, 734
- entry point, 744, 750, 757, 758
- file
  - extension, appending to archive file, 791
  - linking, 748, 750, 789, 800–803, 868
- flow control, 767–768, 790–791
- Generic Conduit Base Classes, using
  - application info block synchronization, 789
  - category synchronization, 766
  - Conduit Wizard, selecting in, 744
  - flow control, 767–768
  - format, describing to conduit, 768–770
  - introduced, 730–731
  - link converter, 789
  - opening conduit, 754, 756
  - record format, converting between desktop and handheld, 779, 780, 782, 785–786, 788
  - record retrieval, 766, 770–776
  - record storage, 766, 770, 776–779
  - Sync Manager API, interaction with, 731

- information about, returning, 750, 751–753
  - installing
    - automatic, 735–738
    - manual, 732–735
  - Librarian application
    - CPalmRecord instance, converting data to/from, 782, 785–786, 788
    - retrieving desktop record, 773–776
    - storing desktop record, 777–779
    - structure approach, 768, 770
  - link converter class, 789, 800, 872
  - location, 734
  - MFC base class, using
    - category synchronization, 789
    - Conduit Wizard, selecting in, 744
    - determining if conduit is MFC-based, 750, 753
    - flow control, 790–791
    - generic conduit versus, 731
    - hierarchy, 790
    - introduced, 730
    - link converter class, 800, 872
    - monitor, 789, 791–792
    - opening conduit, 757, 790–791
    - portability considerations, 731
    - record format, converting between desktop and handheld, 789, 791, 800–803
    - record storage in table, 790, 791, 792–796
  - mirror image synchronization environment, 730, 747
  - name
    - display name, assigning, 734
    - returning, 750, 753–754
  - one-directional synchronization environment, 730
  - opening, 750, 754–757, 767, 790–791
  - organizer upload/download, 747
  - plug-in module, as, 726
  - portability, 731
  - priority, 734
  - project, 743, 748, 749
  - record
    - converting format using Generic Conduit Base Classes, 779, 780, 782, 785–786, 788
    - converting format using MFC base class, 789, 791, 800–803
    - describing format to conduit, 768–770
    - retrieval, 766, 770–776
  - registration
    - Conduit Manager, using, 736–738
    - displaying registered conduit, 732
    - editing registration entry, 735
    - HotSync, using, 728, 731
    - InstallShield, using, 735
    - key, primary, 756, 761
    - manual, 732, 734
    - path, 756, 761
    - Sync Manager, using, 804
    - unregistering, 732, 735, 738, 804
  - schema class, 879
  - structure, handling desktop data using, 768, 770
  - table-based, 744
  - transaction-based synchronization environment, 729
  - uninstalling, 735, 738
  - version, returning, 750, 754
- Conduit Configuration tool, 732–735**
  - Conduit Development Kit. See CDK**
  - Conduit Manager tool, 731, 735–738**
  - Conduit Switch tool, 735**
  - Conduit Wizard, 743–750, 790, 792, 793**
  - ConduitCfgEnum enumeration, 761
  - ConfigureConduit function, 750, 758–760, 761
  - ConfigureSubscription function, 750
  - confirmation dialog box alert, 19**
  - Connection Management Protocol (CMP), 675**
  - connector, universal, 567. See also serial communication**
  - console mode debugging, 141–142**
  - Constructor utility**
    - alert editor window, 157–158
    - alignment, 172
    - App Icon Family editor window, 161
    - app info string editor window, 160
    - Arrange Objects dialog box, 187
    - catalog window, 171, 861
    - Create New Resource dialog box, 156
    - form layout window, 169–170, 185, 868
    - hierarchy window, 170–171, 870
    - menu bar editor window, 172
    - menu editor window, 174
    - PRC file, opening, 153
    - project, 152, 153–157
    - string list editor window, 159
    - version covered in this book, 152
    - Wave Sound editor window, 163
  - ConstructRecord method, 791
  - control. See also specific control**
    - defined, 863
    - event handling, 211
    - font, 836
    - form, adding to dynamic, 841, 842
    - group, 219–220, 864
    - style, 842

# ControlStyleType enumeration

---

- ControlStyleType **enumeration, 842**
  - ConvertFromRemote **function, 800, 802–803**
  - ConvertGenericToPc **method, 779–780**
  - ConvertPcToGeneric **method, 774, 779, 780–781**
  - ConvertToRemote **function, 800, 801–802**
  - coordinate system, 340, 342–343, 345, 874, 882**
  - Copilot software, 864**
  - count **field, 619**
  - CountDatabases **function, 480**
  - CPalmRec.h **file, 782**
  - CPalmRecord **class**
    - converting data to/from CPalmRecord instance, 779, 780, 782, 785–786, 788
    - GetRawData method, 780
    - hierarchy, 767
    - introduced, 766
  - CPCategory **class, 766, 767, 768**
  - CPCategoryMgr **class, 766, 767, 768**
  - CPcMgr **class**
    - AddCRs method, 780
    - AddRec method, 773
    - Close method, 772
    - CloseDB method, 772
    - ConvertGenericToPc method, 779
    - ConvertPcToGeneric method, 779
    - CreateDBInfo method, 772
    - hierarchy, 767
    - introduced, 766
    - Open method, 772
    - OpenDB method, 772, 777
    - ReadInData method, 773, 775
    - RetrieveDB method, 770, 771, 773
    - StoreDB method, 770
    - SynchronizeAppInfo method, 789
    - WriteOutData method, 777, 779
  - CPDbBaseMgr **class, 766, 767**
  - CPString **class, 769**
  - CRawRecordInfo **class, 800–801, 808–809, 810**
  - Create New Resource dialog box (Constructor), 156**
  - CreateDBInfo **method, 772**
  - CreateDirectory **function, 581–584, 585**
  - CreateFile **function, 586–589**
  - CreateTable **method, 791**
  - creationDate **field, 471**
  - creator **field, 471**
  - creator ID**
    - case sensitivity, 53
    - database, 53, 474, 479, 734
    - described, 53, 864
    - directory name, adding to, 584–585
    - entering, 52, 68
    - exchange target Creator ID, 619, 636
    - feature creator, 367
    - FtrGet function creator ID argument, 268
    - HotSync creator ID retrieval, 728
    - launching application, returning when, 300, 301, 494
    - registering, 52, 53
    - serial hardware identifier, 678, 679
    - STRT value, 52, 68
  - cryptography, public key, 877**
  - CSS (Cascading Style Sheets), 280, 284**
  - CString **class, 769**
  - CStringField **class, 798**
  - CSynchronizer **class, 766, 767, 768, 773**
  - CSyncPreference **class, 760–761**
  - CSyncProperties **class, 755–756, 772**
  - ctdbuild **file, 64**
  - ctdproject **file, 64**
  - ctlEnterEvent **object, 211, 213**
  - ctlExitEvent **object, 212, 213**
  - CtlGetLabel **function, 220**
  - CtlGetValue **function, 217, 219**
  - CtlGlueGetFont **function, 218**
  - CtlGlueSetFont **function, 836**
  - CtlHandleEvent **function, 211**
  - CtlNewControl **function, 841, 842**
  - CtlNewGraphicControl **function, 841, 842**
  - CtlNewSliderControl **function, 841**
  - ctlRepeatEvent **object, 212, 213**
  - ctlSelectEvent **object, 211, 213, 220, 245, 543**
  - CtlSetLabel **function, 220, 221, 242, 246**
  - CtlSetValue **function, 219**
  - CtlValidatePointer **function, 843**
  - CTS (clear to send) signal, 674, 695, 696, 697**
  - cursor**
    - color, 336
    - record reference, 550–553, 864
  - Custom dialog box (HotSync), 751, 758**
  - CustomReceiveDialog **function, 652–654**
  - customTableItem **style, 389, 390, 391, 406, 421**
- ## D
- DAL (Device Abstraction Layer), 134, 864**
  - data link protocol, 704**
  - data terminal ready (DTR) signal, 674**
  - data type**
    - database type, specifying, 474
    - exchange manager, registering with, 620, 621–626
    - wrapping object in object of different type, 623

**database. See also field; record; table**

access mode, 476–477, 481, 490  
 application info block, 485–488, 543, 789, 859  
 backing up, 68, 481, 483, 729  
 binding, 552–553  
 classic, 470, 476, 488, 862  
 closing, 477, 490, 807  
 conduit, identifying to, 734  
 creating, 473–476, 488, 489–490  
 creator ID, 53, 474, 479, 734  
 data type, specifying, 474  
 debugging, 122–123, 136, 137–139  
 defined, 864  
 deleting, 53, 480–481  
 exchanging, 667–671  
 existence, checking, 474  
 expansion card, transferring to/from, 615  
 exporting to file, 615  
 extended, 261, 470, 488, 614  
 file stream, 482  
 finding on handheld, 477–480  
 header, 471, 483, 486, 518–519, 557  
 Hello World 1 application, 81  
 importing from file, 615  
 information about, returning, 478, 481–485, 532  
 memory  
   chunk, 258, 261  
   handle, 487–488  
   RAM-resident database, 479  
 modification date, 483  
 module database, 81  
 multiple databases, working with, 474  
 naming, 474  
 number of databases on card, returning, 478  
 opening, 476–477, 490, 532, 804–807  
 Palm OS Simulator, working with in, 136, 137–139  
 PODS, 68  
 POSE, installing in, 122–123  
 protection count, 480, 877  
 Protein environment, 476, 478  
 read-only, 476, 481, 490  
 read/write, 476, 490  
 record, cutting and pasting to another database, 525  
 referencing, 81, 490  
 resource database, 17, 200, 201, 473, 481  
 schema, 261, 432, 470, 488–492, 549–555  
 sharing, 490  
 size, returning, 485  
 sort info block, 882  
 storage, in secondary, 614–615

system preference database, 495, 496  
 timestamp, 483  
 version, 482, 531–536  
 visibility, 482  
 write-only, 476, 490

**datagram, UDP, 704**DataMgr.h file, **482, 507, 537****date**

calculation involving, 369–371  
 database modification date, 483  
 file date information, 596–597  
 formatting, 223, 371–374  
 picker, 221–223, 864  
 record sorting by, 537  
 table, inserting in, 389, 390–391, 398–399  
 user preference, 371–374

**Date Book dialog box (Librarian), 204**DateAdjust function, **371**DateDaysToDate function, **369**DateFormatType enumeration, **370**DateSecondsToDate function, **369**dateTableWidgetItem style, **389, 390–391**DateTime.h file, **367, 370–371, 373, 390**DateTimeType structure, **369**DateToAscii function, **369–370**DateToDOWDMFormat function, **223, 370**DateToInt macro, **399**DateType structure, **367–368, 369****Daylight Saving Time (DST), 372–374**DayOfMonth function, **370**DayOfMonthType enumeration, **370–371**DayOfWeekType enumeration, **371**DaysInMonth function, **371**DbAddCategory function, **555**DbAddColumn function, **492**DbAddSortIndex function, **554–555**DbAddTable function, **490**dbCardNo field, **663**DbCloseDatabase function, **490**DbCopyColumnValue function, **552**DbCopyColumnValues function, **552**DbCreateDatabase function, **489, 490**DbCursorBindData function, **552–553**DbCursorClose function, **551**DbCursorMove functions, **552**DbCursorOpen function, **550–551**DbCursorSetAbsolutePosition function, **552**DbDeleteRow function, **554**DbEnableSorting function, **555**

# DbGetAllColumnDefinitions function

---

- DbGetAllColumnDefinitions **function, 492**
- DbGetCategory **function, 555**
- DbGetColumnDefinitions **function, 492**
- DbGetColumnID **function, 492, 552**
- DbGetColumnValue **functions, 552**
- DbGetSortDefinition **function, 555**
- DbGetTableName **function, 492**
- DbGetTableSchema **function, 492**
- DbHasSortIndex **function, 555**
- dbID **field, 663**
- DbInsertRow **function, 553–554**
- DbNumCategory **function, 555**
- DbNumColumns **function, 492**
- DbNumSortIndices **function, 555**
- DbNumTables **function, 492**
- DbOpenDatabase **function, 490**
- DbOpenDatabaseByName **function, 490**
- DbReleaseStorage **function, 552**
- DbRemoveCategory **function, 555**
- DbRemoveColumn **function, 492**
- DbRemoveRow **function, 554**
- DbRemoveSortIndex **function, 555**
- DbRemoveTable **function, 492**
- DbSchemaColumnDefnType **structure, 491**
- DbSetCategory **function, 555**
- DbTableDefinitionType **structure, 489, 490–491**
- Debug ↔ Kill (CodeWarrior), 143**
- Debug Options dialog box (POSE), 125**
- Debug ↔ Step Into (CodeWarrior), 143**
- Debug ↔ Step Out (CodeWarrior), 143**
- Debug ↔ Step Over (CodeWarrior), 143**
- debugging**
  - battery, simulating low, 118
  - breakpoint, 140–141, 143
  - console mode debugging, 141–142
  - database application, 122–123, 136, 137–139
  - error, sending execution to external debugger upon, 133
  - event, viewing, 139
  - exchanging data, 624, 666–667
  - expansion card, emulating, 132–133
  - gadget, 28
  - Graffiti debugging shortcut, 146–147, 666–667
  - Gremlin, using
    - Palm OS Simulator, 136–137
    - POSE, 28, 127–129, 134, 869, 882
  - input, using random, 126
  - IR communication, 117, 124
  - memory
    - emulating available, 122
    - heap, viewing, 138
    - snapshot, saving/loading, 139
  - pointer, validating, 843
  - resetting handheld, 145–146
  - ROM image, 121–122, 134–135, 337, 879
  - serial communication, 124
  - stepping through code, 143, 145
  - synchronization, 125, 130–132
  - tracing, 883
- define **statement, 148**
- Delete Book dialog box (Librarian), 193**
- delete **field, 472**
- DeleteAlert **resource, 202**
- DeleteDirectory **function, 589–593**
- DeleteFile **function, 589, 590–591, 593, 595**
- DeleteNote **function, 531**
- DeleteRecord **function, 519–520**
- description **field, 620, 626–627**
- Desktop Link Protocol (DLP), 674, 675**
- destination (Multimedia Services), 353, 354**
- DetailsDeleteRecord **function, 204–206**
- Device Abstraction Layer (DAL), 134, 864**
- DeviceInfoType **structure, 693**
- dialog box. See also specific dialog box**
  - alarm, displaying in response to, 364–365, 366
  - alert
    - button, 158, 166, 200–201
    - color, 336
    - confirmation, 19
    - creating, 157–158, 165–166
    - displaying, 199–200
    - error, 19
    - file data, displaying in, 603
    - Help ID value, 158, 165
    - information-only, 18
    - message text, 158, 166, 201–202
    - resource database reference, 200, 201
    - text field, displaying using, 95–97
    - title bar, 158, 166
    - type, assigning, 158, 165
    - warning, 19
  - alignment, 36
  - attention manager
    - detail dialog box, 374–375
    - interrupting current operation, 377
    - list dialog box, 375, 379, 384
    - modal dialog box, 374

- border, 182
- closing, 206–207, 366
- conduit, 758, 759
- control value, returning, 204
- defined, 865
- event handling, 203
- exchange manager progress, 620
- exchanging data, displaying acceptance dialog box
  - when, 649–654, 663
- Help ID value, 181
- initializing, 206
- label, 36
- modal, 181–182, 202, 203–207, 374, 873
- Save Behind flag, 181
- sizing, 182
- spacing, 36
- user interface guidelines, 35–36
- digitizer, 865**
- directory**
  - access mode, 594, 595
  - application, 569
  - backup location, 569
  - content, enumerating, 611–612
  - creating, 581–585
  - default, 613, 865
  - deleting, 589–593
  - locking, 594
  - naming, 584
  - path, 584, 593
  - root, 569
  - structure, recommended, 569
- dirty bit, 516, 521, 525, 537, 865**
- dirty field, 472**
- DiscoverSchema method, 795**
- DisplayAlarm function, 365, 366**
- DLL (Dynamic Link Library), 726, 728, 734, 757**
- DLP (Desktop Link Protocol), 674, 675**
- DmArchiveRecord function, 518**
- DmAttachRecord function, 514, 516–518, 523–528**
- DmCloseDatabase function, 477**
- DmCompareF function, 507–508**
- DmCreateDatabase function, 473–474, 476, 488, 532**
- DmCreateDatabaseV50 function, 476**
- DmDatabaseInfo function, 531–532**
- DmDatabaseInfoType function, 478, 481, 487**
- DmDatabaseProtect function, 480–481**
- DmDatabaseSize function, 485**
- DmDeleteCategory function, 548**
- DmDeleteDatabase function, 480**
- DmDeleteRecord function, 518**
- DmDetachRecord function, 519**
- DmFindDatabase function, 477–478, 490**
- DmFindDatabaseByTypeCreator function, 300, 360, 376**
- DmFindDatabaseV50 function, 478**
- DmFindRecordByID function, 514**
- DmFindRecordByOffsetInCategory function, 430**
- DmFindSortPosition function, 513, 515, 517, 522, 663**
- DmGetDatabase function, 478**
- DmGetDatabaseLockState function, 484**
- DmGetDatabaseV50 function, 478**
- DmGetLastError function, 478, 549**
- DmGetNextDatabaseByTypeCreator function, 301, 360, 376, 479–480**
- DmGetNextDatabaseByTypeCreatorV50 function, 480**
- DmGetOpenInfo function, 484**
- DmGetRecord function, 520–522, 525, 536**
- DmGetResource function, 333, 334, 352**
- DmInsertionSort function, 515, 536, 537**
- dmModeShowSecret mode, 548**
- DmMoveCategory function, 546**
- DmMoveRecord function, 518–519, 521, 522, 523, 663**
- DmNewHandle function, 487–488, 516, 523**
- DmNewRecord function, 514, 515–516, 517, 663**
- DmNumDatabases function, 478**
- DmNumRecords function, 536**
- DmNumRecordsInCategory function, 536, 542, 548**
- DmOpenDatabase function, 476**
- DmOpenDatabaseByTypeCreator function, 476**
- DmOpenDatabaseInfo function, 484, 487**
- DmOpenDatabaseV50 function, 476**
- DmOpenIteratorByTypeCreator function, 480**
- DmOpenRef reference, 200**
- DmPositionInCategory function, 542**
- DmQueryNextInCategory function, 542, 548**
- DmQueryRecord function, 520, 663**
- DmQuickSort function, 515, 537**
- DmRecordInfo function, 537, 543, 563**
- DmReleaseRecord function, 515, 516, 521, 529**
- DmRemoveRecord function, 518**
- DmRemoveSecretRecords function, 549**
- DmResizeRecord function, 536, 549**
- DmSearchRecord function, 514**
- DmSearchStateType structure, 479**
- DmSeekRecordInCategory function, 430, 540–542, 548**

DmSet **function, 497**

DmSetDatabaseInfo **function, 483**

DmSetRecordInfo **function, 538**

DmStrCopy **function, 497, 520**

DmWrite **function, 497, 516, 521, 536**

**domain, network, 709**

**dotted decimal notation, IP address, 708, 711, 865**

**doze mode, 12, 84, 865**

**DragonBall processor, 678, 865**

DrawBitmap **function, 333–334**

DrawCharsInWidth **function, 276–279**

DrawCustomTableItem **function, 408–409**

**drawing**

anti-aliasing, 344–345

arc, 347

attention manager draw request, 382–383, 384

bitmap, 327, 333–334

circle, 347

clipping region, 347, 862

database record, 438, 439–440

Draw Detail request, 382

fill pattern, 329, 330–332, 868

form, 86, 88, 93, 179–180, 208

gadget, 234–236, 239, 240

graphic, 326–332, 345

header string, 557

icon, 392, 406

line, 328–329, 345

list, 242–245, 383

path, 344, 346–347, 875

pixel, 328

rectangle, 329–330, 332, 346

rendering state, 345–346

routine, custom, 864

state, 318–319, 327, 342, 865

table

cell, 390, 440

column, 440

item, 389, 391, 406–408

redrawing, 410, 413, 459, 463

row, 405, 410, 432, 438–440

text, 274, 275, 279, 332–333, 347

transfer mode, 326–327, 884

transformation, 345

window, 312–313, 315, 318–319, 866

DrawNarrowTextTableItem **function, 408**

DrawRecordName **function, 439**

**DST (Daylight Saving Time), 372–374**

**DTR (data terminal ready) signal, 674**

**Dynamic Link Library (DLL), 726, 728, 734, 757**

## E

**Eclipse Workbench**

Additional Project Settings screen, 69–70

compiling using, 62, 74

Editor, 71

linking, 62, 74

MCP file support, 65

Outline view, 71–72

perspective, 72

project

C/C++ project, 71, 74

creating, 64–68

displaying, 70–72

editing, 73–74

file, adding, 72–73

generating project code, 68–69

opening, 64

PNO project, 64

Protein project, 64

rebuilding, 74

Properties dialog box, 73–74

Task view, 72, 74

workspace, 63

**Edit ⇄ Clear Item Text (Constructor), 185**

**Edit ⇄ Clear Menu Item (Constructor), 173**

**Edit ⇄ Clear Object (Constructor), 172**

**Edit ⇄ Clear (PODS), 177**

**Edit ⇄ Clear Resource (Constructor), 157**

**Edit ⇄ Clear String (Constructor), 159**

**Edit ⇄ Copy Object (Constructor), 171**

**Edit ⇄ Copy (PODS), 177**

**Edit ⇄ Duplicate Object (Constructor), 172**

**Edit ⇄ Edit Resource (Constructor), 156**

**Edit menu, importance of providing standard, 32**

**Edit ⇄ New Column Width (Constructor), 189**

**Edit ⇄ New Family Element (Constructor), 161**

**Edit ⇄ New Item Text (Constructor), 185**

**Edit ⇄ New Menu Bar Resource (Constructor), 172**

**Edit ⇄ New Menu (Constructor), 172**

**Edit ⇄ New Menu Item (Constructor), 173**

**Edit ⇄ New Resource ⇄ Menu Bar (PODS), 178**

**Edit ⇄ New Resource (PODS), 176**

**Edit ⇄ New Separator Item (Constructor), 173**

**Edit ⇄ New String (Constructor), 159**

**Edit ⇄ New Type Resource (Constructor), 156**

**Edit ⇄ Paste Object (Constructor), 171**

**Edit ⇄ Paste (PODS), 177**

**Edit ⇄ Remove Menu (Constructor), 173**

**Edit With ⇄ Resource Editor (PODS), 164**

**Edit With ⇄ Text Editor (PODS), 164**

- EditFormGetFieldHeight **function, 450, 452**
- EditFormGetRecordField **function, 467**
- EditFormHandleEvent **function, 250–253, 459, 464**
- EditFormInit **function, 441–442**
- EditFormLoadTable **function, 442, 449–452**
- EditFormResizeData **function, 464–467**
- EditFormSaveRecordField **function, 461–463**
- EditFormScroll **function, 455–459, 463**
- EditFormUpdateScrollers **function, 459, 461**
- EditInitTableRow **function, 451, 453–454**
- EditSetGraffitiMode **function, 467**
- 8-bit color, 319–320**
- ellipsis character, drawing, 279**
- enter event handling, 211, 213, 409–410, 866**
- ErrDisplay **macro, 147, 148**
- ErrFatalDisplayIf **macro, 147, 148**
- ErrNonFatalDisplayIf **macro, 147, 148**
- error dialog box alert, 19**
- error handling**
  - Berkeley sockets, 706, 707
  - compilation, during, 147–148
  - database find failure, 478
  - debugger, sending execution to external upon error, 133
  - directory operation, 584, 585, 593
  - exchange manager socket connection, 641, 642
  - file operation, 593, 610, 851, 854–855, 857
  - HotSync, 739, 740
  - macro, displaying error using, 147, 148
  - network
    - library error, 706, 707, 708
    - receive error, 719
    - send error, 718
    - socket connection, 717
  - out of range error, 828
  - POSE, 133–134
  - record resize failure, 549
  - serial communication
    - break signal error, 691
    - CD signal error, 691
    - CRC-16 error checking, 674
    - framing error, 691
    - handshake error, 691
    - hardware overrun, 691
    - line error, 691, 700
    - parity error, 691
    - port closing failure, 683
    - port opening failure, 679, 680, 699
    - receiving data, 690, 691
    - sending data, 683, 685
    - serial manager existence check failure, 676–677
    - software overrun, 691
    - synchronization, 739, 740
    - VFS, 585–586
- error manager API, 147–148**
- eSyncPref **enumeration, 761**
- eSyncRecAttrs **enumeration, 809**
- eSyncTypes **enumeration, 761**
- event defined, 866**
- event handling. See also specific event**
  - alarm, 365–366
  - battery, low, 306
  - command toolbar, 249
  - control event, 211
  - date picker, 222–223
  - debugging, viewing event when, 139
  - dialog box, 203
  - enter event, 211, 213, 409–410, 866
  - exit event, 211–212, 213, 867
  - expansion card event, 573
  - form event
    - close, 203, 206, 233
    - dispatch, 83, 85
    - enter, 211, 213
    - exit, 211–212, 213
    - go to, 564
    - load, 81, 84, 564
    - open, 81, 88, 93, 233–234, 392
    - save, 557
    - update, 88, 181, 208–209, 344
  - gadget, 236–239, 241
  - list, 211, 212, 213, 244
  - loop, 16–17, 82, 83, 294–295, 685–687
  - menu event, 84, 93–95, 211, 247
  - palmOne 5-way navigation button, 814–815
  - pen event, 236–237, 241, 289–292, 876
  - pop-up, 211, 213, 245–246
  - queue, 866
  - repeat event, 878
  - scrolling, 211, 212
  - select event, 211, 213, 220, 880
  - serial communication, 685–687
  - Sony CLIE, 817–819, 820–821, 822, 823, 824
  - table, 211, 212, 213, 409–410, 463–464
  - volume event, 573, 575
- EventLoop **routine, 685–687**
- EvtAddEventToQueue **function, 299**
- EvtEventAvail **function, 685**

# EvtGetEvent function

---

`EvtGetEvent` **function, 83, 685**

**exchange library**

- Bluetooth, 638, 639
- defined, 867
- existence, determining, 639–640
- identifying, 619, 642
- setting, 620, 637–640
- URL, 637–640

**exchange manager**

- data type, registering, 620, 621–626
- file
  - extension, registering, 622, 623
  - socket, identifying to, 620, 621, 636
- MIME type, working with, 620, 622–623, 636
- progress dialog box, 620
- socket connection
  - closing, 641, 642
  - communication socket versus, 618
  - description, 626–627
  - error handling, 641, 642
  - `ExgAccept` function, passing to, 620
  - file, identifying to, 620, 621, 636
  - initializing, 626–627, 636–640
  - MIME type, identifying to, 620, 636
  - number of objects in, setting, 619
  - opening, 640–641, 659
  - pointer, 640, 641, 654, 659
  - referencing, 619

**exchanging data**

- connectionless packet mode, 619
- debugging, 624, 666–667
- error handling, 641, 642
- launching application
  - acceptance dialog box, displaying at launch time, 649–650
  - bug in launcher, 671
  - completion, upon, 619, 621, 654
  - data type, registering, 623
- length of data transferred
  - returning, 641
  - setting, 619, 640
- local, 619, 637, 640
- Memo Pad application, 640
- operating system version, checking, 654
- previewing, 620
- progress dialog box, 620
- receiving
  - acceptance dialog box, 649–654, 663
  - beginning transfer, 659
  - buffer size considerations, 663

- category, entire, 665
  - database, entire, 670–671
  - ending transfer, 664
  - launching application when, 654
  - looping to retrieve incoming data, 659–663, 665
  - multiple records, 665
  - resizing record when, 663
- record
- category, receiving entire, 665
  - category, specifying, 650–654, 663
  - displaying, 640, 664, 666
  - multiple records, receiving, 665
  - multiple records, sending, 643
  - resizing record when receiving, 663
  - sort position, determining, 663
- remote, 619

**sending**

- application, entire, 667–670
  - application on same handheld, to, 617, 637, 640
  - beginning transfer, 640–641
  - Bluetooth device, to, 638, 639–640
  - database, entire, 667–670
  - ending transfer, 642
  - exchange library, specifying, 637–639
  - initializing connection, 626–627, 636–640
  - length of data transferred, setting, 640
  - looping to send outgoing data, 641
  - records, multiple, 643
  - target, specifying, 636
- 68K application, 620
- target
- Creator ID, 619, 636
  - socket initialization, specifying in, 636

Ultra mode, 619

`ExgAccept` **function, 620, 659**

`ExgAskParamType` **structure, 649–650**

`ExgDBRead` **function, 670–671**

`ExgDBWrite` **function, 667–668**

`ExgDialogInfoType` **structure, 654**

`ExgDisconnect` **function, 641, 642, 664**

`ExgDoDialog` **function, 650, 654**

`ExgGoToType` **structure, 664**

`ExgMgr.h` **file**

`ExgAskParamType` structure definition, 649–650

`exgBeamPrefix` constant, 638

`exgBeamScheme` constant, 638

`ExgDialogInfoType` structure definition, 654

`ExgGoToType` structure definition, 664

`exgLocalPrefix` constant, 638

`exgReg` constants, 622

exgSendBeamPrefix constant, 638  
 exgSendPrefix constant, 638  
 exgSendScheme constant, 638  
 introduced, 618  
**ExgPut function, 640–641, 642**  
**ExgReceive function, 659, 663, 665**  
**ExgRegisterData function, 624**  
**ExgRegisterDatatype function, 620–626**  
**ExgSend function, 641–642**  
**ExgSocketType structure**  
   acceptedSocket field, 620  
   appData field, 619, 663  
   cnvFrom68KApp field, 620  
   componentIndex field, 620  
   count field, 619  
   description field, 620, 626–627  
   goToCreator field, 619, 664  
   goToParams field, 619, 664  
   header file, 618  
   initializing, 626–627, 636–640  
   length field, 619, 640  
   libraryRef field, 619, 642  
   localMode field, 619, 640, 666  
   name field, 620, 622, 636, 637  
   noGoTo field, 619, 640  
   noStatus field, 620  
   packetMode field, 619  
   padding\_1 field, 620  
   preview field, 620  
   reserved field, 620  
   socketRef field, 619  
   target field, 619, 621, 622, 636  
   time field, 619  
   type field, 471, 473, 620, 636  
**exit event handling, 211–212, 213, 867**  
**expansion card**  
   application  
     launching application resident on expansion card, 570–571  
     launching upon inserting, 569–570  
   database, transferring to/from, 615  
   emulating, 132–133  
   event handling, 573  
   file stored on, accessing, 354  
   flash memory, 569, 573  
   insertion, detecting, 572–573  
   memory module card versus, 256  
   notification, 573–576  
   removal, detecting, 572–573

  volume, 568  
   writing data to both RAM and expansion card, 599  
**expansion manager, 567–568**  
**expansion slot, 867**  
**exporting data using data exchange. See exchanging data**

## F

**FAT (File Allocation Table), 568, 584**  
**feature**  
   backlight, 272  
   checking for specific, 271–273, 286  
   creator, 267, 867  
   defined, 867  
   index value, 272–273  
   International Feature Set, 286  
   memory, 497–498, 867  
   number, 267, 867  
   pointer, 268  
   set, 867  
   table, 268, 867  
   value, returning, 268, 273  
   version, 268  
**feedback slider, 867**  
**FEP (Front-End Processor), 868**  
**field. See also specific field**  
   attribute, setting, 231–232  
   bytes contained, returning, 97  
   change, checking for, 463  
   database header, representation in, 471  
   defined, 868  
   Editable property, 183  
   focus  
     returning, 218  
     setting, 32, 93, 230–231, 232  
     table, 463, 467  
     text field, 868  
   Graffiti shortcut, 33–34, 184, 230  
   input field, creating, 89, 183  
   iteration, 450, 459  
   justification, 25  
   length  
     returning, 229  
     setting maximum, 183  
   memory  
     design considerations, 226–227  
     handle, text field, 227, 229, 393–394, 405, 463  
     offset, 503, 533

## field (continued)

---

### field (continued)

- multiline, 25, 26, 183, 874
- pointer, 88, 96–97, 218
- record sorting, checking for key field change when, 527–528
- rows, spanning multiple, 450
- scrolling, 184, 440–441, 458–459
- selection in, replacing, 229
- single-line, 25, 26, 183, 881
- sizing, 183, 400, 441–442, 459
- sorting, 439
- table
  - focus, 463, 467
  - handle, 393–394, 463
  - label, 391
  - note, 399
  - referencing, 461
  - spacing, 401, 452
  - uniqueness of associated field object, 461
- text
  - capitalization, automatic, 183
  - color, 336
  - copying, 308
  - deleting, 229
  - edit mode, 463, 866
  - focus, 868
  - font, 391, 836
  - form, adding to dynamic, 841
  - handle, 227, 229, 393–394, 405, 463
  - label, 391
  - modifying, 228–229
  - offset, 533
  - returning, 97, 229–230
  - underlining, 183, 336
- tracking field containing data, 503, 506
- `FieldAttrType` **structure**, 231–232
- `Field.h` **file**, 231–232
- file**
  - access mode, 594, 595, 610, 852
  - alert, displaying file data in, 603
  - attribute, returning, 595
  - closing, 589, 593–594
  - conduit
    - extension, appending to archive file, 791
    - file linking, 748, 750, 789, 800–803, 868
  - creating, 586–589
  - database, exporting to/importing from, 615
  - date information, 596–597
  - deleting, 589–593
  - end of file, determining, 605–606, 857
  - error handling file operation, 593, 610, 851, 854–855, 857
  - exchange manager
    - extension, registering, 622, 623
    - socket, identifying file to, 620, 621, 636
  - finding, 610–612
  - font definition file, 837
  - location, returning default, 613–614
  - locking, 594
  - management, 569–570, 571–572
  - name, conversion of long, 584
  - opening, 593–594, 604
  - pointer, 603–606, 868
  - reading, 598–603, 604, 856–858
  - size, returning, 597–598
  - stream
    - access mode, 852
    - closing, 854
    - defined, 868
    - deleting, 854, 855
    - end of file, determining, 857
    - error handling, 851, 854–855, 857
    - opening, 851–854
    - overhead, 850
    - position in, setting, 855–856
    - reading, 856–858
    - rewinding, 856
    - speed, 850–851
    - truncating, 855
    - writing, 853, 858
  - system support, 569
  - writing, 599, 604, 607–610, 853, 858
- File Allocation Table (FAT)**, 568, 584
- File** ⇄ **Import (Eclipse Workbench)**, 73
- File** ⇄ **New (Conduit Wizard)**, 743
- File** ⇄ **New** ⇄ **File (Eclipse Workbench)**, 72
- File** ⇄ **New Project**
  - CodeWarrior, 48
  - Eclipse Workbench, 65
- File** ⇄ **New Project File (Constructor)**, 152
- File** ⇄ **Open Project File (Constructor)**, 153
- File** ⇄ **Save a Copy As**
  - CodeWarrior, 58
  - Constructor, 171
- File** ⇄ **Save (Constructor)**, 153
- file streaming API**, 850–858
- `FileClearerr` **function**, 855
- `FileClearError` **function**, 851
- `FileClose` **function**, 851, 854

- FileDelete **function, 855**
- FileDmRead **function, 857–858**
- FileEOF **function, 851, 857**
- FileError **function, 851, 854–855**
- FileGetLastError **function, 854–855**
- FileInfoType **structure, 611**
- FileOpen **function, 851–854**
- FileRead **function, 851, 856–857**
- FileRewind **function, 851, 856**
- FileSeek **function, 851, 855–856**
- FileTell **function, 851, 856**
- FileWrite **function, 851, 858**
- fill pattern, 329, 330–332, 868**
- FindDrawHeader **function, 557**
- FindSaveMatch **function, 557**
- FindStrInStr **function, 557, 561, 844**
- firstEntry **field, 472**
- 5-way navigation button programming, palmOne, 813–816, 859**
- flash memory, 569, 573**
- FldCalcFieldHeight **function, 459**
- fldChangedEvent **object, 183, 184**
- FldCompactText **function, 529**
- FldCopy **function, 308**
- FldCut **function, 308**
- FldDelete **function, 229**
- FldDrawField **function, 227, 228**
- fldEnterEvent **object, 211, 213**
- FldGetAttributes **function, 232**
- FldGetTextHandle **function, 227, 229, 529**
- FldGetTextLength **function, 97, 229**
- FldGetTextPtr **function, 97, 229**
- fldHeightChangedEvent **object, 463–464**
- FldInsert **function, 229**
- FldNewField **function, 841**
- FldPaste **function, 308**
- FldSetAttributes **function, 232**
- FldSetFocus **function, 232**
- FldSetFont **function, 836**
- FldSetSelection **function, 229**
- FldSetText **function, 520**
- FldSetTextHandle **function, 227, 529**
- FlipWord **function, 803**
- FntAverageCharWidth **function, 274**
- FntBaseLine **function, 274**
- FntCharHeight **function, 275**
- FntCharsInWidth **function, 276, 279**
- FntCharsWidth **function, 275**
- FntCharWidth **function, 275, 424**
- FntDefineFont **function, 835, 837**
- FntDescenderHeight **function, 275**
- FntGetFont **function, 274**
- FntLineHeight **function, 275, 429**
- FntLineWidth **function, 275**
- FntSetFont **function, 274, 332**
- font**
  - alternate, 281
  - application preference, storing, 493
  - ascender, 282, 837
  - baseline, 274, 860
  - bitmap, 273–275, 282, 838
  - bold, 280, 422
  - bounding box, returning, 282
  - button, 218
  - color, 226, 327–328, 336
  - control, 836
  - CSS, formatting using, 280, 284
  - current, setting/returning, 274, 282
  - custom
    - adding in Garnet environment, 835–836
    - creating, 836–839
  - definition file, 837
  - descender, 275, 282, 837, 864
  - determining available, 283–284
  - drawing, 274, 275, 279, 332–333, 347
  - face, 280
  - family, 280–281, 282, 283
  - field, text, 391, 836
  - fitting within specified width, 276–279, 283
  - form, 180
  - handle, 281–282, 283
  - hinting, 345
  - ID, 274
  - italic, 280
  - Librarian application, 194, 421–422, 493
  - list, 836
  - memory, releasing, 282
  - oblique, 280
  - resource, 835
  - scalable, 274, 279–284
  - size
    - returning, 274–275, 282, 424
    - setting, 280–281, 282, 837
  - style, 280, 282
  - table, 389, 391, 421–422
  - TrueType, 274, 281, 835
- for **loop, 398–400**

**form.** See also *specific form element*

- active, 86, 366, 859
- adding object, 177
- alignment, 172
- bitmap, 184, 868
- border, 181
- closing, 203, 206, 207, 563, 842
- color, 336
- comment, 179
- copying object, 171, 172, 177
- defined, 868
- deleting object, 172, 177
- drawing, 86, 88, 93, 179–180, 208
- ease of use, designing for, 31
- event handling
  - close, 203, 206, 233
  - dispatch, 83, 85
  - enter, 211, 213
  - exit, 211–212, 213
  - go to, 564
  - load, 81, 84, 564
  - open, 81, 88, 93, 233–234, 392
  - save, 557
  - update, 88, 181, 208–209, 344
- font, 180
- Graffiti shortcut, 33–34, 184, 230
- hiding object, 180, 215–217
- ID, 840
- index value of object in, 96, 206, 214
- initializing, 204, 233, 840
- label, 184–185, 841
- loading, 81
- mandatory, 18
- menu bar, 18
- modal, 181, 214
- moving object, 177
- Note form, 193
- Object Identifier, 179
- origin property, 179
- palmOne 5-way navigation button, designing for, 816
- pointer, 88, 96–97, 213–215, 218
- record category, displaying in, 208
- saving, 181, 556–557
- sizing, 180
- switching to new, 202, 203
- threading, 86
- title bar, 18, 31, 840, 883
- updating, 88, 181, 208–210, 344
- FormGadgetTypeInCallback **structure**, 240–241

**frame**

- button, 182, 868
- rectangle, 878
- window, 313–314, 315, 868
- FrmAlert **function**, 199–200
- FrmCloseAllForms **function**, 82, 563
- frmCloseEvent **object**, 203, 206, 233, 236
- FrmCopyLabel **function**, 184
- FrmCustomAlert **function**, 96, 201, 202, 585
- FrmDeleteForm **function**, 95, 204, 366
- FrmDispatchEvent **handler**, 83, 85
- FrmDoDialog **function**, 95, 202, 203–206, 214
- FrmDrawForm **function**, 88, 93, 234
- FrmGetActiveField **function**, 218
- FrmGetActiveForm **function**, 88, 214
- FrmGetControlGroupSelection **function**, 219–220
- FrmGetControlValue **function**, 204
- FrmGetFocus **function**, 230
- FrmGetFormPtr **function**, 88
- FrmGetGadgetData **function**, 236
- FrmGetObjectIndex **function**, 96, 206, 214, 215, 842
- FrmGetObjectIndexFromPtr **function**, 215
- FrmGetObjectPtr **function**, 96, 213, 214
- FrmGetWindowHandle **function**, 88
- FrmGlueGetActiveField **function**, 218
- frmGotoEvent **function**, 564
- FrmGotoForm **function**, 81, 202, 203
- FrmHandleEvent **function**, 208, 210, 211
- FrmHideObject **function**, 215, 839
- FrmInitForm **function**, 86, 95, 204, 206, 840
- frmLoadEvent **object**, 81, 84, 564
- FrmNewBitmap **function**, 841
- FrmNewForm **function**, 839–840
- FrmNewGadget **function**, 841
- FrmNewGsi **function**, 841
- FrmNewLabel **function**, 841
- frmOpenEvent **object**, 81, 88, 93, 233–234, 392
- FrmPopupForm **function**, 202, 206–207
- FrmRemoveObject **function**, 842–843
- FrmReturnToForm **function**, 207
- FrmSaveAllForms **function**, 556–557
- frmSaveEvent **object**, 557
- FrmSetActiveForm **function**, 86, 366
- FrmSetControlGroupSelection **function**, 219–220
- FrmSetControlValue **function**, 204, 206
- FrmSetEventHandler **function**, 87
- FrmSetFocus **function**, 32, 93, 230
- FrmSetGadgetHandler **function**, 239–240

**FrmShowObject function, 215, 217, 839**  
**frmTitleEnterEvent object, 211**  
**frmTitleSelectEvent object, 211**  
**frmUpdate**  
   event, 181  
   structure, 208  
**frmUpdateEvent object, 88, 208–209, 344**  
**FrmUpdateForm function, 208**  
**FrmUpdateScrollers function, 461**  
**FrmValidatePtr function, 843**  
**Front-End Processor (FEP), 868**  
**FtrGet function**  
   application number argument, 268  
   character encoding, determining using, 849–850  
   creator ID argument, 268  
   exchange library existence, determining using,  
     639–640  
   feature  
     memory, returning value from using, 497  
     pointer, returning using, 268  
   international manager existence, determining using,  
     843  
   operating system version, determining using, 268  
   palmOne 5-way navigation button, detecting using,  
     815–816  
   processor ID, determining using, 271  
   serial manager support, determining using, 676  
   Sony CLIÉ Jog Dial existence, determining using, 819  
   Sony CLIÉ JogAssist pointer, returning using, 823  
   VFS manager existence, determining using, 572  
**FtrGetByIndex function, 272–273**  
**FtrPtrFree function, 498**  
**FtrPtrNew function, 497**  
**FtrPtrResize function, 498**  
**function**  
   callback, 861  
   calling in Garnet environment, 13

## G

### gadget

  boundary, returning, 27–28, 236  
   closing, 236, 240–241  
   debugging, 28  
   defined, 869  
   drawing, 234–236, 239, 240  
   event handling, 236–239, 241  
   extended, 239–241, 867  
   form, adding to dynamic, 841  
   initializing, 233

  pointer, 28, 233  
   state, storing in variable, 233  
   value, returning current, 236  
**Garnet Palm OS**  
   ARM support, 12–13  
   Berkeley sockets support, 703, 722  
   font, adding custom, 835–836  
   function, calling, 13  
   header file, 111, 113  
   kernel, 14–15  
   memory  
     architecture, 255–259  
     reference, 16-bit, 827  
   PalmOSGlue library support, 13  
   PNO support, 13  
   processor, 12–13  
   32K jump limit, 827, 828–830  
   variable, considerations when using global, 576

**GcBeginClip function, 347**  
**GcClosePath function, 346**  
**GcColor function, 345**  
**GcCountFontFamilies function, 283**  
**GcCountFontStyles function, 283–284**  
**GcCreateFont function, 279–280**  
**GcCreateFontFromFamily function, 281, 283**  
**GcCreateFontFromID function, 282**  
**GcDrawTextAt function, 347**  
**GcEndClip function, 347**  
**GcFontStringCharsInWidth function, 283**  
**GcFontStrongBytesInWidth function, 283**  
**GcGetCurrentContext function, 345**  
**GcGetFontBoundingBox function, 282**  
**GcGetFontFamily function, 283**  
**GcGetFontFamilyAndStyle function, 282**  
**GcGetFontHeight function, 282**  
**GcGetFontSize function, 282**  
**GcGetFontStyle function, 283–284**  
**GcInitGradient function, 345**  
**GcLineTo function, 344, 346, 347**  
**GcMoveTo function, 344, 346, 347**  
**GcPaint function, 344, 346, 347**  
**GcPopState function, 345, 347**  
**GcPushState function, 345, 347**  
**GcRect function, 344**  
**GcReleaseContext function, 345**  
**GcReleaseFont function, 282**  
**GcSetAntialiasing function, 345**  
**GcSetFont function, 282, 345**  
**GcSetFontAntialiasing function, 345**

# GcSetFontHinting function

---

GcSetFontHinting **function, 345**

GcSetFontSize **function, 281**

GcSetFontTransform **function, 345**

GcSetGradient **function, 345**

GcSetJoin **function, 345**

GcSetPenSize **function, 345**

GcStroke **function, 344**

GcTransform **function, 345**

**General Packet Radio Service (GPRS), 869**

**Generic Conduit Base Classes. See also specific class**

application info block synchronization, 789

category synchronization, 766

Conduit Wizard, selecting in, 744

flow control, 767–768

format, describing to conduit, 768–770

hierarchy, 767

introduced, 730–731

link converter, 789

opening conduit, 754, 756

record

format, converting between desktop and handheld,

779, 780, 782, 785–786, 788

retrieval, 766, 770–776

storage, 766, 770, 776–779

Sync Manager API, interaction with, 731

GetBitMacro **macro, 507**

GetConduitInfo **function, 750, 751–753**

GetConduitName **function, 750, 752, 753–754**

GetConduitVersion **function, 750, 752, 754**

GetData **method, 797**

GetIPAddress **function, 712–713**

GetObjectPtr **function, 215**

GetRawData **method, 780**

GetServicePort **function, 714–715**

GetTextColumn **function, 402–403**

**Global System for Mobile (GSM) communication, 870**

**Go There request, 383**

**Got It request, 384**

goToCreator **field, 619, 664**

GoToItem **function, 562–563, 564**

goToParams **field, 619, 664**

**GPRS (General Packet Radio Service), 869**

**gradient, color, 345**

**Graffiti shortcut**

area, 869

caps-lock mode, 293

case-shift mode, 293

command stroke, 863

command toolbar, 19–20, 248

debugging shortcut, 146–147, 666–667

form, 33–34, 184, 230

keyDownEvent object result argument, storing in,  
250

menu item, assigning to, 173, 174, 179

POSE input, 117, 118, 119

shift indicator, 26, 32, 184, 841, 869

speed, 29

**graphic. See also bitmap; icon**

anti-aliasing, 344–345

backward compatibility, 311–312

clipping region, 347, 862

Cobalt graphics model, 343–347

context, 343–344, 345, 869

drawing, 326–332, 345

Palm Image Checker tool, 875

**Gremlin tool**

Palm OS Simulator, 136–137

POSE, 28, 127–129, 134, 869, 882

Gremlin\_nnn\_Events\_Min.txt **file, 129**

Gremlin\_Root\_State.psf **file, 128**

**Gremlins dialog box (Palm OS Simulator), 136**

**Gremlins ⇄ Minimize (POSE), 129**

**Gremlins ⇄ New (POSE), 127**

**Gremlins ⇄ Replay (POSE), 129**

**GSM (Global System for Mobile) communication, 870**

## H

HandleCommonMenus **command, 638**

HandleNotification **routine, 575–576**

**Handspring Visor memory card, 255–256**

**hard reset, 870**

**header, database, 471, 483, 486, 518–519, 557**

**header file**

Berkeley sockets, 706

CDK

bfields.h, 798

Condapi.h, 753, 761

CPalmRec.h, 782

eDbOpenModes constant, 804

pcmgr.h, 772

syncmgr.h, 753, 755, 760, 800

tables.h, 798

Cobalt, 111, 113

include directive, 78–79

resource header file, 155

resource ID constant definition, creating, 175

Sony CLIÉ, 817–818, 819

version 3.5 differences, 111–113

**Hello World 1 application**

AppStart routine, 81  
 AppStop routine, 82  
 closing, 82  
 code listing, complete, 97–102  
 database, 81  
 EvtGetEvent function, 83  
 FrmCloseAllForms routine, 82  
 FrmGetFormPtr function, 88  
 FrmGetWindowHandle function, 88  
 include directive, 78–79  
 launch code, 80–81  
 MenuHandleEvent handler, 84  
 PilotMain function, 79–80  
 PrvInvalidateFormWindow function, 88  
 SysTimeToSecs routine, 83

**Hello World 2 application**

about dialog box, 89–90, 95  
 AboutForm resource, 95  
 code listing, complete, 102–110  
 input text field, 89  
 MainMenuHandleEvent handler, 93–96  
 PrvInvalidateFormWindow function, 93  
 SaySomething function, 93, 95–96

helloMain.c **file**, 78

helloResources.h **file**, 79, 90–91

**hexadecimal**

IP address format, 709  
 number value, 268, 285

**High-Density Display feature set, 340–341****host byte order, 712, 870**

hostent **structure**, 710

**HostFS utility, 132–133****hostname resolution, 709–711, 712****HotSync tool. See also synchronization**

CondSwitch command, 735  
 configuration, backing up, 735  
 creator ID retrieval, 728  
 Custom dialog box, 751, 758  
 error handling, 739, 740  
 FastSync option, 728, 867  
 introduced, 2  
 logging, 738–741, 791, 792  
 Manager, 130, 131, 726, 727, 735  
 notification, 726, 728, 874  
 PC ID value, 728, 876  
 POSE, emulating in, 125, 130–132  
 power considerations, 8  
 record, deleting, 554  
 SlowSync option, 728

speed, 729  
 starting, 727  
 user validation, 728  
 Windows Registry, configuring using, 728, 731

**I****IANA (Internet Assigned Numbers Authority), 623****icon**

bitmap, 25, 162–163, 166  
 color, 160–161, 162, 167, 326  
 compression, 162  
 defined, 870  
 density, 161, 162, 167  
 drawing, 392, 406  
 family, 160–161, 871  
 form tips icon, 18, 883  
 grayscale, 161, 162  
 large, 871  
 launcher, 36  
 multibit, 873  
 resource icon, assigning, 154, 155, 160–162  
 sizing, 161, 163, 167  
 small, 881  
 table, inserting in, 391, 392, 399, 406, 411  
 transparency, 162

id **field**, 473

IDD\_CONDUIT\_ACTION **dialog box**, 758

**IDE (integrated development environment), 45–46, 870.**

*See also specific IDE*

**image. See graphic**

ImportData **function**, 750

**importing data using data exchange. See exchanging data**

**importing database from file, 615**

include **directive**, 78–79

**increment arrow, 871**

**info block**

application, 485–488, 543, 789, 859  
 sort, 882

**information dialog box alert, 18**

**Infrared Data Association (IrDA), 38, 871**

**infrared (IR) communication, 38, 117, 124, 696, 871**

.ini **files**, 492

InitVFS **function**, 578, 579

**in-place editing, 528–531, 871**

**insertion point, 25, 871**

**InstallShield software, 735**

**integrated development environment (IDE), 45–46, 870.**

*See also specific IDE*

**Intel little-endian byte order, 803**  
**interface, user. See user interface**  
**International Feature Set, 286, 287–288, 871**  
**international manager feature, 843**  
**Internet Assigned Numbers Authority (IANA), 623**  
**Introduction to Palm OS Developer Suite, 74**  
**IP (Internet Protocol) address, 708–712, 865**  
**IR (infrared) communication, 38, 117, 124, 696, 871**  
**IrDA (Infrared Data Association), 38, 871**  
**IsDigit macro, 286, 844**  
**IsPrivate method, 797**

## J

**Jog Dial programming (Sony CLIÉ), 816–819, 871**  
**JogAssist programming (Sony CLIÉ), 819–826**  
**jump island, 830**

## K

**kAttnCommandDrawDetail action, 382**  
**kAttnCommandGoThere action, 383**  
**kernel, 14–16**  
**keyDown event, 814, 820**  
**keyDownEvent object**  
application button, resulting from pressing, 288, 294  
battery, triggered in reaction to low, 306  
chr member, 292  
field focus move, 230  
Graffiti-related, 250, 292  
launch constant, 250, 299  
modifiers member, 293  
result argument, 250  
testing if press occurred while handheld turned off, 296  
testing incoming character from, 287  
virtual, 293, 294  
**keyHold event, 814**  
**keyUp event, 814**

## L

### label

button, 31, 182, 186, 188, 218  
check box, 23  
defined, 871  
dialog box, 36  
field, text, 391  
form, 184–185, 841  
justification, 36  
table, 389, 391, 393

trigger  
pop-up, 186, 246  
selector, 24, 188–189, 220–221  
volume, 579–580, 584, 595  
**labelNoColonTableItem style, 390, 392, 421**  
**labelTableItem style, 389, 391, 421**  
**language, localizing application for, 286**  
**LANSync Preferences dialog box (POSE), 131**  
**largeBoldFont function, 422, 429**  
**lastBackupDate field, 471**  
**launch code defined, 872**  
**launching**  
alarm, 360, 362–363, 364, 366  
application launcher, 298–299, 860  
attention manager launch code response, 375, 376, 378–381  
creating custom launch code, 302  
creator ID of application launched, returning, 300, 301, 494  
exchanging data  
acceptance dialog box, displaying at launch time, 649–650  
bug in launcher, 671  
completion, launching application upon, 619, 621, 654  
data type, registering at launch time, 623  
receiving data, upon, 654  
expansion card  
application, launching upon inserting, 569–570  
application resident on, launching, 570–571  
Hello World 1 application, 80–81  
**keyDownEvent object launch constant, 250, 299**  
normal, 16  
operating system version requirement, checking, 301–302  
parameter, passing to application, 300  
POSE, application in, 123  
preference, loading when, 493–494  
process, sublaunching, 15, 16  
quitting running application before launching new, 300  
sending launch code to application, 299–302  
table, initializing at application launch, 392–393  
**Layout ⇄ Hide Object IDs (Constructor), 170**  
**Layout ⇄ Show Object Hierarchy (Constructor), 170**  
**.lbc files, 624, 643**  
**LED (Light Emitting Diode), flashing using attention manager, 377**  
**length field, 619, 640**

- .lib files, **624**
- LibAppInfoInit **function, 485–486, 487–488, 539**
- LibAppInfoType **structure, 486**
- LibChangeRecord **function, 525–527**
- LibComparePackedRecords **function, 508–511, 513, 514**
- LibConvertDB **function, 534–536**
- LibDBRecordFlags **structure, 503, 527**
- LibDBRecordType **structure, 502, 504**
- LibFindKey **function, 511–513**
- LibFindSortPosition **function, 513–514, 663**
- LibGetAppInfo **function, 449**
- LibGetDatabase **function, 474–476, 483–484, 485, 490, 533–534**
- LibGetRecord **function, 438–439, 449, 459**
- LibGetSortOrder **function, 439**
- LibImportRecord **function, 659–663**
- LibNewRecord **function, 517–518**
- LibPackedDBRecord **structure, 502, 503, 532, 533**
- LibPackedDBRecordV0 **structure, 532–533**
- LibPreferenceType **structure, 492–493, 494**
- Librarian application**
  - About Librarian dialog box, 194, 195
  - Beam Book option, 196, 248
  - Beam Category option, 194
  - Book Details dialog box, 192–193, 544
  - Change Security dialog box, 194
  - conduit
    - CPalmRecord instance, converting data to/from, 782, 785–786, 788
    - retrieving desktop record, 773–776
    - storing desktop record, 777–779
    - structure approach, 768, 770
  - Date Book dialog box, 204
  - Delete Book dialog box, 193
  - Edit menu, 196
  - Edit view, 192–193, 196, 414–415, 440–441, 442
  - exchange library URL scheme, 638
  - exchange manager, interaction with
    - acceptance dialog box display, 649–654, 663
    - data type registration, 624–626
    - sending data, 641–642
    - socket initialization, 626–627
  - feature overview, 190
  - font, 194, 421–422, 493
  - List view
    - category display, 191
    - introduced, 190
    - menu bar, 194
    - New button, 191
    - screen width, fitting text to, 276–279
    - Show button, 191
    - updating, 209–210
  - Note view, 193, 529–530
  - Options menu, 194
  - PilotMain function
    - category selection prompt coding, 650–652
    - CustomReceiveDialog function call, 652
    - LibRegisterData function call, 624
    - multi-segment application, 556
    - sysAppLaunchCmdFind routine handling, 558
    - sysAppLaunchCmdGoto routine handling, 561–562
    - sysAppLaunchCmdReceiveData routine handling, 655
    - sysAppLaunchCmdSaveData routine handling, 557
    - sysAppLaunchFlagNewGlobals flag, 561
  - preference, storing, 492–493
  - Preferences dialog box, 191
  - Record menus, 194, 196
  - Record view, 191–192, 196
  - screen density, 340, 341
  - Send Book option, 196
  - Send Category option, 194
- librarian.c **file, 492**
- librarianDB.c **file, 528**
- librarianDB.h **file, 486, 502**
- librarian.h **file, 415**
- LibrarianRecord **structure, 770, 773, 779, 872**
- librarianTransfer.h **file, 625**
- libraryRef **field, 619, 642**
- LibReceiveData **function, 664–665**
- LibRegisterData **function, 624, 625**
- LibSendCategory **function, 643–649**
- LibSendRecord **function, 627, 636, 638–639**
- LibStatusType **structure, 502**
- LibUnpackedSize **function, 503–504**
- Light Emitting Diode (LED), flashing using attention manager, 377**
- line, drawing, 328–329, 345**
- link converter class, 789, 800, 872**
- list**
  - attention manager list dialog box, 375, 379, 384
  - defined, 872
  - drawing, 242–245, 383
  - dynamic, 185, 243–245
  - event handling, 211, 212, 213, 244
  - font, 836
  - form, adding to dynamic, 841

## list (continued)

---

### list (continued)

- item
  - adding, 185, 377
  - defined, 872
  - number of items in list, returning, 242
  - removing, 185
  - selection, 241, 242, 880
  - text, returning, 241–242
  - visibility, 185, 242
- placement, dynamic, 185
- pointer, 241–242
- pop-up
  - category selector, 186, 191, 192, 547, 555
  - defined, 876
  - event handling, 245–246
  - speed, 29
  - table item, calling from, 389, 391
  - trigger, 21, 185, 186, 245–246
- record list structure, 472
- scrolling, 21, 185
- sizing, 185
- table versus, 387
- text
  - passing string array to, 243
  - string list, 159–160, 859, 882
- ListFormDrawRecord **function, 432, 439–440**
- ListFormHandleEvent **handler, 209, 288, 415, 546**
- ListFormInit **function, 415–422**
- ListFormLoadTable **function, 422, 425–428, 429–432**
- ListFormNextCategory **function, 288, 546–547**
- ListFormNumberOfRows **function, 428–429**
- ListFormUpdateDisplay **routine, 209–210**
- ListFormUpdateScrollButtons **function, 432**
- little-endian byte order, 712, 803, 872**
- load routine, custom, 864**
- LoadTextTableItem **function, 402, 403**
- localChunkID **field, 472, 473**
- LocalID memory offset, 470–472, 478, 479, 537, 872**
- localizing application**
  - International Feature Set, 286, 287–288, 871
  - international manager, using, 843
  - language, 286
  - text manager, using, 843–844
- localMode **field, 619, 640, 666**
- lock count, 872**
- LogAddEntry **function, 738–739**
- LogApplicationName **method, 791**
- LogRecordData
  - function, 792
  - method, 791

- LogTestCounters **function, 740–741**
- looping**
  - event loop, 16–17, 82, 83, 294–295, 685–687
  - exchanging data
    - incoming data, looping to retrieve, 659–663, 665
    - outgoing data, looping to send, 641
  - file end, determining, 605–606
  - for loop, 398–400
  - serial communication
    - local loopback test, 696
    - receive event loop, 685–687
  - while loop, 432, 450, 511
- LstDrawList **function, 242–243**
- lstEnterEvent **object, 211, 213**
- LstEraseList **function, 242–243**
- lstExitEvent **object, 212, 213**
- LstGetNumberOfItems **function, 242**
- LstGetSelection **function, 241**
- LstGetSelectionText **function, 241–242, 244, 246–247**
- LstGetVisibleItems **function, 242**
- LstGlueSetFont **function, 836**
- LstHandleEvent **function, 211**
- LstMakeItemVisible **function, 242**
- LstNewList **function, 841**
- lstSelectEvent **object, 211, 213, 241**
- LstSetDrawFunction **function, 243–245, 247**
- LstSetListChoices **function, 243, 244, 246**
- LstSetSelection **function, 242**

## M

- MainFormEventHandler **routine, 410**
- MainFormHandleEvent **function, 87–88, 91–93, 291, 392, 409–410**
- MainFormInit **function, 233, 392, 394–398**
- MainFormSelectColorDepth **function, 322–323**
- MainFormSwitchPalette **function, 325**
- MainListDraw **function, 245**
- MainMenuHandleEvent **handler, 84, 93–96**
- Make **command (CodeWarrior), 61**
- makefile, PODS, 65–66, 834**
  - .map **files, 832**
- matchCustom **field, 663**
- MCP files, 51, 65**
- MemHandleFree **function, 262–263**
- MemHandleLock **function, 229, 261, 333, 352**
- MemHandleNew **function, 261**
- MemHandleResize **function, 263**
- MemHandleSize **function, 403**
- MemHandleUnlock **function, 228**

MemInitHeapTable **API, 264**

MemKernelInit **API, 264**

MemLocalIDToLocketPtr **function, 488**

MemLocalIDToPtr **function, 264**

MemMove **function, 263**

**Memo Pad application data exchange, 640**

## memory

Background Process, 16

card

address space, 256, 259

AlmSetAlarm function cardNo parameter, 360

AttnGetAttention function cardNo parameter,  
376

databases on card, returning number of, 478

Dm functions cardNo parameter, 473–474, 483

expansion card versus, 256

LocalID, 470–472, 478, 479, 537, 872

logical, 255–256

SysAppLaunch function cardNo parameter, 300

SysGetROMToken function cardNo parameter, 306

SysNotifyRegister function cardNo parameter,  
574

chunk

database, 258, 261

described, 256, 862

record, 258, 469, 516–517, 518, 523

Cobalt memory architecture, 259–261, 264

comparing values in, 264

database

chunk, 258, 261

RAM-resident database, 479

desktop compared, 5–6

dynamic RAM, 256–258, 259–261, 264, 866

emulating available memory when debugging, 122

feature memory, 497–498, 867

field

design considerations, 226–227

handle, text field, 227, 229, 393–394, 405, 463

offset, 503, 533

flash memory, 569, 573

fragmentation, 259, 868

garbage collection, 262

Garnet Palm OS

architecture, 255–259

reference, 16-bit, 827

handle

audio, 352, 353

bitmap, 333

database, 487–488

defined, 870

field, text, 227, 229, 393–394, 405, 463

font, 281–282, 283

memory pointer, 261–262

record, 514, 519, 529, 536

heap

dynamic, 256–258, 866

storage, 258, 259, 882

viewing, 138

importing data, buffer size considerations when, 663

locking, 261, 262, 463, 872

moving content, 263

NetHostInfoBufType structure buffer, 709–710

network

receive buffer, 719

socket buffer, 717, 718

TCP/IP stack, 257

palmOne, 258

pointer, 261–262

POSE memory available, specifying, 122

RAM

database, RAM-resident, 479

defined, 877

dynamic, 256–258, 259–261, 264, 866

storage, 256–258, 259–261, 568, 570, 882

writing data to both RAM and expansion card, 599

record

chunk, 258, 469, 516–517, 518, 523

handle, 514, 519, 529, 536

offset, 470

releasing, 515, 516, 521, 529

resizing, 263

resource

memory-resident, 17

stack space, assigning, 156

ROM, 879

serial communication

receive buffer, 694, 700, 880

send buffer, 696, 880

setting value at specified location, 263

snapshot, saving/loading, 139

Sony CLIÉ, 258

Springboard memory module, 255–256

TCP/IP stack usage, 257

text

handle, text field, 227, 229, 393–394, 405, 463

iterating through text buffer, 847–848

returning character from text buffer, 847–849

transliterating character in text buffer, 845

word bounds, returning, 848–849

## memory (continued)

---

### memory (continued)

- UART buffer, 674
- variable, allocating for, 233
- window
  - buffering, double, 316, 865
  - removing from memory, 314

**Memory Stick, 570**

`memorymgrcompatibility.h` file, **264**

`MemoryMgr.h` file, **264**

`MemPtrFree` function, **261, 264**

`MemPtrNew` function, **233, 261–262, 264, 709–710**

`MemPtrResize` function, **263**

`MemPtrSetOwner` function, **300**

`MemPtrUnlock` function, **262, 488**

`MemSet` function, **263, 626**

**menu**

- bar, 20, 172, 173, 175–176, 178
- color, 335, 337
- command shortcut, 19, 174
- command toolbar, 19–20, 248–253, 873
- creating, 172–175
- defined, 872
- deleting, 173, 175–176
- ease of use, designing for, 32
- event handling, 84, 93–95, 211, 247
- ID, 174, 175, 178
- item
  - adding, 173, 179
  - defined, 873
  - Graffiti shortcut, assigning, 173, 174, 179
  - removing, 173, 179
  - reordering, 173
  - visibility, 247–248
- POSE menu system, 119
- separator bar, 20, 173, 880
- sharing, 175–176, 196
- speed, 29
- title, 172–173, 178

`MenuCmdBarAddButton` function, **249–250**

`menuCmdBarOpenEvent` object, **249, 250, 253**

`MenuHandleEvent` function, **211**

**Message Integrity Check (MIC), 873**

**Metrowerks CodeWarrior IDE. See CodeWarrior IDE**

**MFC (Microsoft Foundation Classes)**

- category synchronization, 789
- Conduit Wizard, selecting in, 744
- determining if conduit is MFC-based, 750, 753
- DLL, 757
- flow control, 790–791
- generic conduit versus, 731

- hierarchy, 790
- introduced, 730
- link converter class, 800, 872
- monitor, 789, 791–792
- opening conduit, 757, 790–791
- portability considerations, 731
- record
  - format, converting between desktop and handheld, 789, 791, 800–803
  - storage in table, 790, 791, 792–796
  - specifying as conduit type, 744
  - version, returning, 753

**MIC (Message Integrity Check), 873**

**Microsoft Foundation Classes. See MFC**

**MIDI (Musical Instrument Digital Interface), 351, 873, 881**

**MIME (Multipurpose Internet Mail Extensions)**

- directory, default, 613, 614
- exchange manager
  - registering with, 622–623
  - socket, identifying to, 620, 636
- IANA registry, 623

`MMSessionAddDefaultTracks` function, **355**

`MMSessionControl` function, **355–356**

`MMSessionSeek` function, **356**

`MMSourceFinalize` function, **355**

**Mobile Geographics Web site, 271**

**modem manager, 674, 675**

**Modem Sync Preferences dialog box (POSE), 130**

`modificationDate` field, **471**

`modificationNumber` field, **471**

**Motorola processor line**

- DragonBall, 678, 865
- 68000 family, 859, 873

**Movie Server (Cobalt), 353**

**MS Autorun software, 570**

**multibit icon, 873**

**Multimedia Services subsystem, 353–357**

**Multipurpose Internet Mail Extensions. See MIME**

**multisection application, 833. See also segment**

**Musical Instrument Digital Interface (MIDI), 351, 873, 881**

`MyCompareFunc` function, **522**

`MyFormHandleEvent` function, **212, 213**

## N

**name** field

- database header, 471
- `ExgSocketType` structure, 620, 622, 636, 637

narrowTableWidgetItem **style**, 389, 392, 398, 406, 408

NetHostInfoBufType **structure**, 709–710, 712

NetHostInfoType **structure**, 710

NetHToNL **macro**, 712, 716

NetHToNS **macro**, 712, 716

NetLibAddrAToIN **function**, 711

NetLibAddrINToA **function**, 711

NetLibClose **function**, 708, 721–722

NetLibDmReceive **function**, 719–720

NetLibFinishCloseWait **function**, 722

NetLibGetHostByAddr **function**, 711

NetLibGetHostByName **function**, 709, 710, 712

NetLibGetServByName **function**, 713–714

NetLibOpen **function**, 707–708

NetLibReceive **function**, 718–719, 720

NetLibSend **function**, 717

NetLibSocketConnect **function**, 716, 717

NetLibSocketOpen **function**, 715

NetMgr.h **file**, 710, 714, 716

NetNToHL **function**, 712

NetNToHS **macro**, 712

NetServInfoBufType **structure**, 714

NetServInfoType **structure**, 714

NetSocketAddrINType **structure**, 716

netSocketAddrRaw **function**, 715

NetSocketAddrType **structure**, 716

NetSocket.c **file**, 707, 722

## network

domain, 709

error handling

library error, 706, 707, 708

receive error, 719

send error, 718

socket connection, 717

hostname resolution, 709–711, 712

IP address, 708–712, 865

layer structure, 704–705

library

Berkeley sockets function, mapping to network library routine, 706

close-wait state, 722, 862

closing, 708, 721–722

error handling, 706, 707, 708

global variable requirement, 706

idle timeout, 706, 707, 721

interface, 705–707

opening, 707–708

referencing, 706, 707

TCP/IP redirection in POSE, 124, 130

memory

receive buffer, 719

socket buffer, 717, 718

TCP/IP stack, 257

port, 713–715

PPP, 704

protocol stack, 704

receiving data, 718–720

record transmission, 720

sending data, 717–718

SLIP, 704

socket

API support, 723

buffer, 717, 718

closing, 721

connecting, 716–717

disconnecting, 720–721

error handling, 717

exchange manager socket versus, 618

number open, maximum, 723

raw, 715

referencing, 716, 717

SSL, 882

TCP/IP, 715

UDP, 715

UNIX compatibility, 722

Sync Manager connection, 811–812

TCP/IP

bandwidth considerations, 704

byte order, 712

memory usage, TCP/IP stack, 257

packet receipt acknowledgement, 704

POSE network library redirection to, 124, 130

socket, 715

support, 37–38

transmission verification, 704

UDP, 704, 715

## New dialog box

CodeWarrior, 48–49, 50

Conduit Wizard, 743

**New File dialog box (Eclipse Workbench), 72**

**New Gremlin Horde dialog box (POSE), 127**

**New Project dialog box (CodeWarrior), 60, 830**

**New Project Information dialog box (Conduit Wizard), 749, 750**

**New Project Wizard (Eclipse Workbench), 66–70**

**New Resource (PODS), 165**

**New Session dialog box (POSE), 120–121, 123**

**New Target dialog box (CodeWarrior), 57, 833**

nextRecordListID **field**, 472

## NFNT resource, 835

noGoTo **field**, 619, 640

no-notification reset, 874

noStatus **field**, 620

NoteViewHandleEvent **function**, 297–298

NoteViewLoadRecord **function**, 529–530

NoteViewSave **function**, 530–531

### notification

expansion card, 573–576

HotSync, 726, 728, 874

NTFullDbg\_enUS.rom **file**, 135

NTFull\_enUS.rom **file**, 135

### number value

character, testing if digit, 286, 844

hexadecimal, 268, 285

random number, generating, 302–303

string, converting to integer, 285

table item, associating integer value with, 389, 391

numericTableItem **style**, 389, 391

numRecords **field**, 472

## O

### object

ID, 874

index, 874

Open **method**, 772

OpenConduit **function**, 750, 754–757, 767, 790–791

OpenDB **method**, 772, 777

OpenSerial **function**, 681, 697–698

Options ⇄ **Info menu**, 36

Options ⇄ **LANSync Prefs (POSE)**, 131

Options menu (Librarian), 194

Options ⇄ **Modem Sync Prefs (POSE)**, 130

Options ⇄ **Phone Lookup (Librarian)**, 297

Options ⇄ **Primary PC Setup (POSE)**, 131

OR operator, 804, 846

ORDER BY **SQL statement**, 551

out of range error, 828

Outgoing **field**, 683

## P

**PACE (Palm Application Compatibility Environment)**, 12–13, 134, 217, 875

packetMode **field**, 619

PackRecord **function**, 504–506, 533

padding\_1 **field**, 620

**PADP (Packet Assembly/Disassembly Protocol)**, 674, 675

pageDownChr **event**, 34

pageUpChr **event**, 34

**Palm Application Compatibility Environment (PACE)**, 12–13, 134, 217, 875

**Palm Image Checker tool**, 875

**Palm Native Object (PNO)**, 13, 64

**Palm OS Application Wizard (CodeWarrior)**, 51–53, 830

**Palm OS Development Suite**. See **PODS**

**Palm OS Emulator**. See **POSE**

**Palm OS Programmer's API Reference**, 218, 263

**Palm OS Protein C/C++ Compiler Tools Guide**, 74

**Palm OS Resource Editor**, 62–63, 163–168, 176–179, 875

### Palm OS Simulator

application, installing, 136

CodeWarrior Debugger, using with, 141

DAL, 134

database, working with, 136, 137–139

Events window, 139

exchanging data, simulating, 624

Gremlin, 136–137

Heaps window, 138

library, 875

memory

heap, viewing, 138

snapshot, 139

PACE support, 134

POSE versus, 117, 135

ROM image, 134–135

starting, 134–135

**Palm Query Application (.pqa) file**, 669

**Palm Simulator**. See **PalmSim**

PalmCompatibility.h **file**, 111

### palmOne handheld

5-way navigation button programming, 813–816, 859

memory, 258

Treo, 813

Tungsten, 267, 270, 813

Zire, 267, 270, 271, 813

**PalmOSGlue library**, 13, 217–219, 286–287, 850

PalmOS.h **file**, 78, 111

PalmOSRuntime\_4i\_A5 **file**, 833

PalmOSRuntime\_2i\_A5.lib **file**, 833

**PalmRez CodeWarrior IDE plug-in**, 46, 59, 60, 875

### PalmSim (Palm Simulator)

CodeWarrior IDE, 47

PODS, 63

PalmSim.exe **file**, 135

palmsim.ini **file**, 135

**PalmSource Web site**

- CDK download, 725
- creator ID, registering, 52, 53
- PODS download, 45
- POSE download, 117
- SDK download, 45
- table manager source code, 387

`PalmTypesCompatibility.h` file, **111, 113**

**password, resetting, 549**

**path**

- conduit Windows Registry, 756, 761
- directory, 584, 593
- drawing, 344, 346–347, 875

**pattern, fill, 329, 330–332, 868**

**PC ID value, 728, 876**

`PceNativeCall` function, **13**

`pcmgr.h` file, **772**

`.pdb` files, **122, 548, 614, 876**

`pdn20.dll` file, **726**

**pen**

- event handling, 236–237, 241, 289–292, 876
- sizing, 345

**Perform method, 768**

`.pev` files, **129**

`PhoneNumberLookup` function, **297, 299**

`Pilot.h` file, **78, 111**

`PilotMain` function

- alarm launch code, 362–363
- Hello World 1 application, 79–80
- introduced, 68
- Librarian application
  - category selection prompt coding, 650–652
  - `CustomReceiveDialog` function call, 652
  - `LibRegisterData` function call, 624
  - multi-segment application, 556
  - `sysAppLaunchCmdFind` routine handling, 558
  - `sysAppLaunchCmdGoto` routine handling, 561–562
  - `sysAppLaunchCmdReceiveData` routine handling, 655
  - `sysAppLaunchCmdSaveData` routine handling, 557
  - `sysAppLaunchFlagNewGlobals` flag, 561
- segmented application, 831, 832
- VFS Test application, 575

**PiIRCodeWarrior plug-in, 47, 539, 836–839**

**pixel, drawing, 328**

**PNO (Palm Native Object), 13, 64**

**PODS (Palm OS Development Suite)**

- code section, 862
- database application development, 68
- Debugger, 63, 143–145
- downloading, 45

Eclipse Workbench

- Additional Project Settings screen, 69–70
- C/C++ project, 71, 74
- compiling using, 62, 74
- creating project, 64–68
- displaying project, 70–72
- editing project, 73–74
- Editor, 71
- file, adding to project, 72–73
- generating project code, 68–69
- linking, 62, 74
- MCP file support, 65
- opening project, 64
- Outline view, 71–72
- perspective, 72
- PNO project, 64
- Properties dialog box, 73–74
- Protein project, 64
- rebuilding project, 74
- Task view, 72, 74
- workspace, 63

library output type, specifying, 67

makefile, 65–66, 834

multisection application, 833

naming application, 67

Palm OS Resource Editor, 62–63, 163–168, 176–179, 875

PalmSim, 63

SDK provided with, 45

segmenting application, 833–834

**Point to Point Protocol (PPP), 704**

**pointer**

- application info block, 488
- audio, 352
- bitmap, 333
- CSS, 280
- exchange manager socket, 640, 641, 654, 659
- feature, 268
- file, 603–606, 868
- form, 88, 96–97, 213–215, 218
- function, 261
- gadget, 28, 233
- IP address, 710
- list, 241–242
- memory, 261–262
- record, 507, 513
- serial port ID, 679
- Sony CLIE JogAssist, 823–824
- table, 389–390, 391
- validating when debugging, 843
- volume, 577

# PointType structure

---

PointType **structure, 236**

popSelectEvent **object, 211, 213, 245–246**

## pop-up

event handling, 211, 213, 245–246

list

category selector, 186, 191, 192, 547, 555

defined, 876

event handling, 245–246

speed, 29

table item, calling from, 389, 391

trigger, 21, 185, 186, 245–246

trigger

anchoring, 186

category, 186, 547

defined, 876

form, adding to dynamic, 841

icon, 21

label, 186, 246

list, 21, 185, 186, 245–246

placement, 22

table column, inserting in, 399

window, 314–316

popupTriggerNoColonTableItem **style, 390, 392**

popupTriggerTableItem **style, 389, 391**

## port

network port, 713–715

serial port

closing, 682–683, 699

configuration, default, 694–695, 697

defined, 880

ID, 679, 876

information about, returning, 691–693, 700

IR connection, enabling/disabling, 696

number, logical, 678, 681, 872

opening, 679–681, 699

pointer, 679

POSE, 124, 130

power considerations, 676, 680

sharing, 679

version 3.3 device or earlier, 700–701

**Porting Applications to Palm OS Cobalt, 219**

**POSE (Palm OS Emulator)**

application

exiting, 123

installing, 122–123

launching, 123

text, copying to, 119–120

audio setup, 124

CodeWarrior Debugger, using with, 141

configuration, saving, 123

database, installing, 122–123

Debug Options dialog box, 125

downloading, 117

error handling, 133–134

exchanging data, emulating, 624, 667

expansion card, emulating, 132–133

feature overview, 115–117

Graffiti input, 117, 118, 119

Gremlin, 28, 127–129, 134, 869, 882

handheld emulated, specifying, 122

HotSync, emulating, 125, 130–132

introduced, 47

IR port setup, 117, 124

keyboard

onscreen, 119

shortcut overview, 118–119

LANSync Preferences dialog box, 131

memory available, specifying, 122

menu system, 119

Modem Sync Preferences dialog box, 130

network library redirection to TCP/IP, 124, 130

New Gremlin Horde dialog box, 127

New Session dialog box, 120–121, 123

Palm OS Simulator versus, 117, 135

PRC file, installing, 122

Primary PC Setup dialog box, 131

Properties dialog box, 123–125, 130, 132

ROM image, 121–122

screenshot, saving, 127, 133

serial port setup, 124, 130

session, saving, 123

sizing onscreen display, 125

skin, 122, 125, 133, 881

speed, 118, 131

starting, 120–121

stopping, 124

top, keeping on, 126

warning, error versus, 133–134

## power

battery

changing battery setting, 304

debugging, simulating low when, 118

event, low battery, 306

information about, returning, 304, 305

keyDownEvent triggered in reaction to low, 306

life expectancy, 5

lithium ion, 11

voltage, 304

warning threshold, 304–305

- external, 305
  - processor considerations, 4
  - screen considerations, 4
  - serial port, 676, 680
  - synchronization considerations, 8
  - PPP (Point to Point Protocol), 704**
  - .pga (Palm Query Application) file, 669**
  - .prc files**
    - CodeWarrior IDE target setup, 60
    - Constructor, opening in, 153
    - introduced, 17
    - POSE, installing in, 122
    - release PRC file, 52
    - storage, in secondary, 614–615
  - pref resource, 156**
  - preference**
    - application preference, 492–495
    - defined, 876
    - system preference, 495, 496–497, 548
    - user preference, 371–374
  - Preferences dialog box (Librarian), 191**
  - PrefGetAppPreferences function, 494–495**
  - PrefGetPreference function, 371–373, 496, 548**
  - PrefGetPreferences function, 496**
  - PrefSetAppPreferences function, 495**
  - PrefSetPreference function, 373, 496–497**
  - preview field, 620**
  - Primary PC Setup dialog box (POSE), 131**
  - process management, 15–16**
  - ProcessAttention function, 381–382**
  - processor**
    - Cobalt, 12
    - DragonBall, 678, 865
    - FEP 868
    - Garnet, 12–13
    - ID, returning, 271
    - Motorola 68000 processor family, 859, 873
    - PACE, 12–13
    - power considerations, 4
    - RISC architecture, 879
  - Programmer's Companion for Sony CLIÉ Handheld, 820**
  - Project ⇄ Add Files (CodeWarrior), 54**
  - Project ⇄ Compile (CodeWarrior), 61**
  - Project ⇄ Create Group (CodeWarrior), 54**
  - Project ⇄ Create New Segment (CodeWarrior), 832**
  - Project ⇄ Create New Target (CodeWarrior), 833**
  - Project ⇄ Create Target (CodeWarrior), 57**
  - Project ⇄ Debug (CodeWarrior), 141**
  - Project ⇄ Disable Debugger (CodeWarrior), 140**
  - Project ⇄ Enable Debugger (CodeWarrior), 140**
  - project file, 64**
  - Project ⇄ Make (CodeWarrior), 61**
  - Project ⇄ Open Project (Eclipse Workbench), 64**
  - Project ⇄ Properties (Eclipse Workbench), 73**
  - Project ⇄ Rebuild Project or Project ⇄ Rebuild All (Eclipse Workbench), 74**
  - Project ⇄ Resume (CodeWarrior), 143**
  - Project ⇄ Run (CodeWarrior), 143**
  - Project ⇄ Set Default Project (CodeWarrior), 62**
  - Project ⇄ Set Default Target (CodeWarrior), 62**
  - Project ⇄ Show Breakpoints (CodeWarrior), 140**
  - Properties dialog box**
    - Eclipse Workbench., 73–74
    - POSE, 123–125, 130, 132
  - protection count, 480, 877**
  - Protein application development platform**
    - choosing as development platform, 13–14
    - database application, 476, 478
    - Eclipse Workbench Protein project, 64
    - serial communication, 674
    - 68K application, porting to, 14
  - PrvInvalidateFormWindow function, 88, 93**
  - .psf files, 123**
  - public key cryptography, 877**
  - push button**
    - aligning, 187
    - bitmap, 186
    - copying, 187
    - defined, 877
    - group, 23, 186, 219–220
    - label, 186
    - sizing, 187
    - speed, 29
    - status
      - returning, 217, 219
      - setting, 187, 219
- ## Q
- question mark (?) exchange manager URL prefix, 637**
- ## R
- RAM (Random Access Memory)**
    - database, RAM-resident, 479
    - defined, 877
    - dynamic, 256–258, 259–261, 264, 866
    - storage, 256–258, 259–261, 568, 570, 882
    - writing data to both RAM and expansion card, 599

## random number, generating, 302–303

.rc files, 492

RectPtInRectangle function, 291

RD (receive data) signal, 674

ReadFile function, 599–603

ReadInData method, 773, 775

Read-Only Memory (ROM), 879

ReadRecord method, 773–774

ReadSerial function, 687–689, 690

ReadString routine, 774–775

ReadToNextTab function, 775–776

receive data (RD) signal, 674

ReceiveData function, 719

## record

appending, 515, 793–795

archiving, 518, 554, 748, 764, 777

attention manager, displaying using, 379

busy bit, 515, 516, 520, 537, 861

category

application info block category information, 485, 486, 488, 543

attributes field category information, 472, 507

current, displaying, 493

cycling, 287–288, 546–547

default, 30, 538

defined, 861

deleting, 545

deleting all records in, 548

editing, 544–545

exchanging data, receiving entire category when, 665

exchanging data, specifying category of incoming record when, 650–654, 663

form, displaying in, 208

initializing, 538–540

masking, 537

membership, enumerating, 555

merging, 545

moving record to another, 546

name, changing, 545

name, deleting, 548

name, returning, 544

pop-up list selector, 186, 191, 192, 547, 555

position, returning, 542

schema database, 555

searching, 430, 540–542, 548

selecting, 543–545

setting, 542–543

sorting by, 507

synchronization, 748, 766, 789

uneditable, 544

user-defined, 33, 485

class, 877

comparison, 507–513, 522, 799

creating, 514–518, 553–554, 663

cursor, 550–553, 864

cutting and pasting to another database, 525

defined, 877

deleting

category, all records in, 548

cursor handling of deleted record, 551

database header, marking record deleted in, 518

HotSync, using, 554

location of deleted record in database, 515,

518–519, 536

private record, 549

search operation, skipping deleted record in, 540

status of deleted record, returning, 538

Sync Manager function, using, 811

synchronization of deleted record, 518, 748, 764

dirty bit, 516, 521, 525, 537, 865

drawing, 438, 439–440

exchanging

category, receiving entire, 665

category, specifying, 650–654, 663

displaying record, 640, 664, 666

multiple records, receiving, 665

multiple records, sending, 643

resizing record when receiving, 663

sort position, determining, 663

extended database, 488

ID, 470–471, 472, 514, 537, 538

index, 430, 432, 438, 459, 554–555

information about, returning, 537–538

iteration, 541, 791, 807–810

linear, 790, 792

list structure, 472

locking, 484

masked, 537, 872

memory

chunk, 258, 469, 516–517, 518, 523

handle, 514, 519, 529, 536

offset, 430, 470

releasing, 515, 516, 521, 529

modifying

DmAttachRecord function, using, 523–528

DmGetRecord function, using, 520–522

FldSetText function, using, 520

string in record, in-place, 528–531, 871

updating, 554

network, transmitting over, 720

number of records, returning, 485, 536

- packed/unpacked, 503–506, 533
- pointer, 507, 513
- private, 476–477, 496, 540, 548–549, 876
- reading, 520, 810
- removing, 518, 554
- schema database, 549–555
- searching
  - binary search, 513
  - category, 430, 540–542, 548
  - global find, 555–556, 561–564, 666, 869
  - handle, by, 514
  - ID, by, 514
  - private record, 548
  - speed, 542
  - string, 557, 561
- secret bit, 537, 538, 548
- serial communication, avoiding receiving partial record in, 690–691
- size
  - maximum, 533
  - returning, 503–504
  - setting, 515, 536, 549, 663
  - variable-length, 502–503
- sorting
  - appending record, sort considerations when, 515
  - category, by, 507
  - comparing records for, 507–508, 512
  - date, by, 537
  - exchanging data, determining sort position when, 663
  - index, 554–555
  - insertion sort, 515, 536, 537, 871
  - key field change, checking for, 527–528
  - name, by, 537
  - position in sort order, adding record at specific, 517
  - position in sort order, changing, 518–519, 521, 522, 523, 663
  - position in sort order, returning, 513–514, 517
  - presorted model, 501–502
  - quick sort, 537
  - speed, 536
- synchronization
  - archiving, 748, 764, 777
  - category, 748, 766, 789
  - changed record, 516, 764–765
  - converting format in generic conduit, 779, 780, 782, 785–786, 788
  - converting format in MFC base class conduit, 789, 791, 800–803
  - deleted, 518, 748, 764
  - describing format to conduit, 768–770
  - matching, 472
  - raw data, 780
  - retrieval, 766, 770–776
  - storage in Generic Conduit Base Classes environment, 766, 770, 776–779
  - storage in MFC base class environment, 790, 791, 792–796
  - tab-delimited record, 770, 773, 775–776
  - table, record storage in during synchronization, 790, 791, 792–796
  - unchanged record, 765
  - table
    - column, loading record title to, 439
    - row, relation to, 415
    - synchronization, record storage in table during, 790, 791, 792–796
    - updating record in, 554
    - title, 439
    - writing, 516, 520, 810–811
- Record ⇨ Delete Item (To Do application), 247**
- `recordList` **field, 471–472**
- `recordNum` **field, 663**
- rectangle**
  - defined, 878
  - drawing, 329–330, 332, 346
  - frame, 878
- `RectangleType` **structure, 236, 313**
- Red-Green-Blue (RGB) color, 323–324, 325–326, 327, 337, 339**
- Reduced Instruction Set Computer (RISC), 879**
- registering conduit**
  - Conduit Manager, using, 736–738
  - displaying registered conduit, 732
  - editing registration entry, 735
  - HotSync, using, 728, 731
  - InstallShield, using, 735
  - key, primary, 756, 761
  - manual, 732, 734
  - path, 756, 761
  - Sync Manager, using, 804
  - unregistering, 732, 735, 738, 804
- Remote Procedure Call (RPC), 674**
- `RemoveBitMacro` **macro, 507**
- rendering state, 345–346. See also drawing**
- repeat event, 878**
- repeating button, 22, 188, 212, 878**
- request to send (RTS) signal, 674, 697**
- `reserved` **field, 620**
- resetting handheld, 145–146, 870, 874, 882**
- resolution, 4, 339–343**

**resource.** See also *specific resource*

- audio, adding, 163
- category, 155, 156, 160
- code resource 64K limit, 827, 830–834
- CodeWarrior project resource, 47, 877
- constant, defining, 90–91
- copying, 171
- database, 17, 200, 201, 473, 481
- defined, 878
- deleting, 157
- editing, 156–157
- font, 835
- fork, 153, 878
- header file, 155
- icon, assigning, 154, 155, 160–162
- ID, 17, 155, 878
- memory
  - resident, 17
  - stack space, assigning, 156
- NFNT, 835
- pref resource, 156
- string resource, 158–160, 166
- system resource, 882
- type, 878
- user interface guidelines, 151–152
- version
  - assigning resource version string, 154
  - targeting Palm OS version, 155–156
- xprf resource, 156

RetrieveDB

- function, 771–776
- method, 770, 771, 773

ReverseField **function, 228**

Rez tool, 878

**RGB (Red-Green-Blue) color, 323–324, 325–326, 327, 337, 339**

RGBColorType **structure, 323–324**

RI (ring indicator) **signal, 674**

RISC (Reduced Instruction Set Computer), 879

ROM image, debug, **121–122, 134–135, 337, 879**

ROM (Read-Only Memory), 879

RomVersionCompatible **function, 301–302**

RPC (Remote Procedure Call), 674

.rsrc files, **153, 163**

RS-232 serial connection, **673, 674, 680, 693, 696**

RTS (request to send) **signal, 674, 697**

running mode, **12, 879**

## S

save routine, **custom, 864**

SaveTextTableItem **function, 404–405**

SaySomething **function, 93, 95–96**

**schema**

- class, 879
- database, 261, 432, 470, 488–492, 549–555
- table, 490–491, 790, 792–793, 795–796

sclEnterEvent **object, 211, 213**

sclExitEvent **object, 212, 213**

SclHandleEvent **function, 211**

sclRepeatEvent **object, 212, 213**

**screen**

- density, 340–342
- desktop compared, 4–5
- fitting text to screen width, 276–279, 283
- power considerations, 4
- resolution, 4, 339–343
- saving screen area to offscreen window, 314
- title bar, 34
- user interface guidelines, 34–36

**scrolling**

- car, 26, 188, 880
- event handling, 211, 212
- field, 184, 440–441, 458–459
- line, 34
- list, 21, 185
- page, 34
- repeating button, 22
- table, 20, 440–441, 454–455, 458–461
- updating scroll bar, 27, 188
- width, scroll bar, 188

**SDK (Software Development Kit)**

- documentation, 46
- downloading, 45
- Palm OS Programmer's API Reference*, 218, 263
- PODS, provided with, 45

Search **function, 558–561**

**searching**

- file, 610–612
- record
  - binary search, 513
  - category, 430, 540–542, 548
  - global find, 555–556, 561–564, 666, 869
  - handle, by, 514
  - ID, by, 514
  - private record, 548
  - speed, 542
  - string, 557, 561

**secret bit, 537, 538, 548**secret **field, 472**Sections.def **file, 833, 834**Sections.h **file, 834****Secure Sockets Layer (SSL), 882**SeekRecord **function, 430, 542****segment**CodeWarrior, segmenting application using, 830–832  
defined, 880

multisegment application, 556, 874

PODS, segmenting application using, 833–834

64K code resource limit, working around via segment-  
ing application, 830–834**Segment Info dialog box (CodeWarrior), 832****select event handling, 211, 213, 220, 880**SELECT **SQL statement, 550, 552, 554–555**SelectDay **function, 221–222**SelectDayType **function, 221****selection**

list item, 241, 242, 880

table item, 880

SelectOneTime **function, 224–225****selector trigger, 23, 24, 188–189, 220–225, 841**SelectTime **function, 223, 224****semicolon (;) exchange manager URL separator, 637**SendData **function, 641–642, 717–718**SendDatabase **function, 668–669**SerClose **function, 699**SerControl **function, 700**SerCtlEnum **enumeration, 700**serDevFtrInfo **field, 693**SerGetSettings **function, 700–701**SerGetStatus **function, 700****Serial Chat application**

feature overview, 677

port

closing, 682–683

number, 678

opening, 680–681

receiving data, 685–689

sending data, 683–684

StartApplication **function, 680–681****serial communication. See also USB (Universal Serial Bus); specific protocol**

background connection, 677–678, 679

baud rate, 679, 680, 681, 695, 700

bits per character configuration, 681

Bluetooth connection, 679

byte-ordering, 674

CD signal, 691

CTS signal, 674, 695, 696, 697

debugging, 124

device connected, returning, 692

driver, virtual, 696

DTR signal, 674

end-of-message character, 690

error handling

break signal error, 691

CD signal error, 691

CRC-16 error checking, 674

framing error, 691

handshake error, 691

hardware overrun, 691

line error, 691, 700

parity error, 691

port closing failure, 683

port opening failure, 679, 680, 699

receiving data, 690, 691

sending data, 683, 685

serial manager existence check failure, 676–677

software overrun, 691

event handling, 685–687

flow control, 695, 697

foreground connection, 677–678, 680–681

hardware configuration

creator ID identifier, 678, 679

flag value, 695–696

handshaking, 695, 698

overrun, 691

introduced, 37

layer structure, 674–675

looping

local loopback test, 696

receive event loop, 685–687

manager

flexibility, 675

layer, 674

multiple connection support, 676

version, 676–677, 698–701

modem manager, 674, 675

multiple connection support, 676

parity, 681, 691, 697

port

closing, 682–683, 699

configuration, default, 694–695, 697

defined, 880

ID, 679, 876

information about, returning, 691–693, 700

IR connection, enabling/disabling, 696

number, logical, 678, 681, 872

### serial communication (continued)

port (**continued**)

- opening, 679–681, 699
- pointer, 679
- POSE, 124, 130
- power considerations, 676, 680
- sharing, 679
- version 3.3 device or earlier, 700–701

Protein application, 674

queue

- content existence, checking for, 689–690
- flushing, 681, 685, 699
- receive queue, 689–690, 694
- transmit queue, 683, 684–685

RD signal, 674

receiving data

- buffer, 694, 700, 880
- end-of-message character, 690
- error handling, 690, 691
- event loop, 685–687
- queue, 689–690, 694
- record, avoiding receiving partial, 690–691
- timeout, 685
- version 3.3 device or earlier, 700

RI signal, 674

RPC, 674

RS-232 connection, 673, 674, 680, 693, 696

RTS signal, 674, 697

sending data, 683–685, 696, 699, 880

SG signal, 673

software overrun, 691, 697

speed, 674

status, returning, 691–692, 700

stop bit, 695, 697

TD signal, 673

timeout, 685, 695, 696, 697, 699

**Serial Link Internet Protocol (SLIP), 704**

**Serial Link Manager (SLM), 674**

**Serial Link Protocol (SLP), 674, 675**

**serial number of handheld, returning, 306–307**

**SerialMgr.h function**

- DeviceInfoType structure definition, 693
- serDevFtrInfo field, 693
- serLineError constants, 691
- serPort constants, 678
- SrmCtlEnum enumeration definition, 695
- srmDefaultCTSTimeout constant, 697
- SrmOpenConfigType structure definition, 679–680
- srmSettings constants, 696–697
- srmStatus constants, 692

SerialMgrOld.h file, 698, 700

SerOpen function, 699

SerReceiveFlush function, 699

SerSend function, 699

SerSend10 function, 699

SerSetReceiveBuffer function, 700

SerSetSettings function, 700–701

SerSettingsType structure, 701

**session**

- Multimedia Services session object, 353–356
- POSE session, saving, 123

SetArchiveFileExt method, 791

SetBitMacro macro, 507

SetData function, 798

**Settings** ⇄ **Debugging (POSE), 125**

**Settings** ⇄ **Error Handling (POSE), 134**

**Settings** ⇄ **Properties (POSE), 123**

**Settings** ⇄ **Skins (POSE), 125**

**SG (signal ground), 673**

**Short Message Service (SMS), 881**

showInList field, 486

**silkscreen button, 32, 881**

**16-bit**

- Garnet memory reference, 827
- port number, 713

**68K application**

- emulating 68K environment, 12–13
- exchanging data, 620
- Motorola processor, 859
- Palm OS Protein, porting to, 14

**64K code resource limit, 827, 830–834**

sizeof7BitChar macro, 639

**slash (/)**

- directory separator, 584
- root directory indicator, 569

**slash, double (//) exchange library URL prefix, 639**

**sleep mode, 12, 362, 860, 881**

**slider**

- bitmap, 189
- color, 225–226
- defined, 881
- feedback, 24, 867, 883
- form, adding to dynamic, 841
- regular, 24, 883
- thumb, 24, 883
- volume control, 24

**SLIP (Serial Link Internet Protocol), 704**

**SLM (Serial Link Manager), 674**

**SLP (Serial Link Protocol), 674, 675**

**Smart code model, 829**

- smartphone, 6–7, 881**
- SMF (Standard MIDI File), 881**
- SMS (Short Message Service), 881**
- SmsLib.h file, **638**
- sndCmdFreqOn **command, 351**
- sndCmdIDType **structure, 351**
- sndCmdNoteOn **command, 351**
- sndCommandType **structure, 351**
- sndDoCmd **function, 350–351**
- sndPlayResource **function, 352–353**
- sndPlaySystemSound **function, 350, 364**
- sndSysBeepType **function, 350**
- socket connection**
  - exchange manager
    - closing, 641, 642
    - description, 626–627
    - error handling, 641, 642
    - ExgAccept function, passing to, 620
    - initializing, 626–627, 636–640
    - MIME type, identifying to, 620, 636
    - network communication socket versus, 618
    - number of objects in, setting, 619
    - opening, 640–641, 659
    - pointer, 640, 641, 654, 659
    - referencing, 619
  - network
    - API support, 723
    - buffer, 717, 718
    - closing, 721
    - connecting, 716–717
    - disconnecting, 720–721
    - error handling, 717
    - exchange manager socket versus, 618
    - number open, maximum, 723
    - raw, 715
    - referencing, 716, 717
    - SSL, 882
    - TCP/IP 715
    - UDP, 715
    - UNIX compatibility, 722
- socketRef **field, 619**
- Software Development Kit. See SDK**
- Sony CLIÉ handheld**
  - event handling, 817–819, 820–821, 822, 823, 824
  - header file, 817–818, 819
  - Jog Dial programming, 816–819, 871
  - JogAssist feature programming, 819–826
  - memory, 258
  - MS Autorun software, 570
  - Palm OS system version number, 270, 271
  - Programmer's Companion for Sony CLIÉ Handheld*, 820
  - SonyChars.h file, **817**
  - SonySysFtrSysInfoType **structure, 819**
  - SonySystemFtr.h file, **819**
  - sort info block, 882**
  - sortInfoID **field, 471**
  - sortKeyFieldBits **macro, 528**
  - SortRecordInfoType **structure, 507**
  - sound. See audio**
  - Sound Manager, 349–353, 882**
  - SoundMgr.h file, **350, 351**
  - source (Multimedia Services), 353, 354**
  - speaker, 349–350**
  - SpecialKeyDown **function, 296–297**
  - speed**
    - database in secondary storage, 615
    - file streaming, 850–851
    - font handle considerations, 282
    - Graffiti, 29
    - HotSync, 729
    - menu, 29
    - pop-up list, 29
    - POSE, 118, 131
    - push button, 29
    - record
      - searching, 542
      - sorting, 536
    - serial communication, 674
    - synchronization, 728, 729, 730, 867, 881
    - text buffer iteration, 848
    - user interface guidelines, 29–30
  - Springboard memory module, 255–256**
  - SQL (Structured Query Language)**
    - data binding, 552–553
    - ORDER BY statement, 551
    - record cursor implementation, 550–551
    - SELECT statement, 550, 552, 554–555
    - WHERE statement, 551
  - SrmClose **function, 680, 682**
  - SrmControl **routine, 695, 700**
  - SrmCtlEnum **enumeration, 695–696**
  - SrmExtOpen **function, 679**
  - SrmExtOpenBackground **function, 679**
  - SrmGetDeviceCount **function, 692**
  - SrmGetDeviceInfo **function, 692**
  - SrmOpen **function, 677–678, 679, 681**
  - SrmOpenBackground **function, 677–678, 679, 693**
  - SrmOpenConfigType **structure, 679–680**
  - SrmReceive **function, 689, 690, 691**
  - SrmReceiveCheck **function, 689–690, 691**
  - SrmReceiveFlush **function, 681**

# SrmReceiveWait function

---

SrmReceiveWait **function, 690, 691**

SrmSend **function, 683**

SrmSendCheck **function, 684–685**

SrmSendFlush **function, 685**

SrmSendWait **function, 683**

SrmSetReceiveBuffer **function, 694**

.ssf **files, 139**

**SSL (Secure Sockets Layer), 882**

**Standard MIDI File (SMF), 881**

StartApplication **function**

application preference, returning, 493–494

database

access mode, setting, 476

existence, checking, 474

version, checking, 532, 533

notification registration, 574

record privacy status, returning, 548

Serial Chat, 680–681

Sony CLIÉ JogAssist configuration, 823, 826

system preference, returning, 496

table initialization, 393–394

VFS Test, 574–575

start.prc **file, 569–570**

StartupCode.lib **file, 833**

stdint.h **file, 111**

stdio **functions, 851**

StopApplication **function, 477, 495**

**storage**

expansion card

application, launching upon inserting, 569–570

application resident on, running, 570–571

database, transferring to/from, 615

detecting insertion/removal, 572–573

emulating, 132–133

event handling, 573

file stored on, accessing, 354

flash memory, 569, 573

memory module card versus, 256

notification, 573–576

volume, 568

writing data to both RAM and expansion card, 599

memory, storage, 256–258, 259–261, 568, 570

primary, 568, 599, 876

secondary, 568–571, 599, 614–615, 880

**Storage ↔ Load (Palm OS Simulator), 139**

**Storage ↔ Save (Palm OS Simulator), 139**

STORAGE\_HEADER **structure, 772**

StoreDB

function, 776–778, 779

method, 770

StrAToI **function, 284**

StrCaselessCompare **function, 285, 513**

StrCat **function, 284**

StrChr **function, 284**

StrCompare **function, 284, 285, 508**

StrCopy **function, 284**

**stream**

file stream

access mode, 852

closing, 854

defined, 868

deleting, 854, 855

end of file, determining, 857

error handling, 851, 854–855, 857

opening, 851–854

overhead, 850

position in, setting, 855–856

reading, 856–858

rewinding, 856

speed, 850–851

truncating, 855

writing, 853, 858

Multimedia Services, 353

StrIToA **function, 285**

StrIToH **function, 285**

StrLen **function, 284**

StrNCaselessCompare **function, 285**

StrNCat **function, 284**

StrNCompare **function, 284, 285**

StrNCopy **function, 284**

StrPrintf **function, 284**

StrStr **function, 284**

**Structured Query Language. See SQL**

StrVPrintf **function, 284**

**stylus tool, 882, 883**

SubscriptionSupported **function, 750**

SwapDWordToIntel **function, 803**

SwapDWordToMotor **function, 803**

switch **statement, 439**

**switching**

application, 300, 339

form, 202, 203

**Sync action configuration dialog box (Conduit Wizard), 748**

**Sync Manager API**

choosing as development environment, 803

conduit, registering/unregistering using, 804

database, opening/closing, 804–807

Generic Conduit Base Classes, interaction with, 731

introduced, 727

- logging, 738
- MFC base class, interaction with, 731
- network connection, 811–812
- record manipulation, 807–811
- SyncCreateDB **function, 805, 807**
- SyncDeleteRec **function, 811**
- synchronization. See also conduit; HotSync tool**
  - application info block, 789
  - carriage return, adding to/removing from text line, 780
  - category, 748, 766, 789
  - component overview, 727
  - debugging, 125, 130–132
  - desktop application, 727, 728, 729
  - error handling, 739, 740
  - fast, 728, 867
  - handheld application, 726, 729
  - information update, 728
  - logging, 738–741, 791, 792
  - logic, native, 763–766, 874
  - mirror image, 730, 747, 763, 873
  - one-directional, 730, 874
  - power considerations, 8
  - record
    - archiving, 748, 764, 777
    - category, 748, 766, 789
    - changed record, 516, 764–765
    - converting format using Generic Conduit Base Classes conduit, 779, 780, 782, 785–786, 788
    - converting format using MFC base class conduit, 789, 791, 800–803
    - deleted record, 518, 748, 764
    - describing format to conduit, 768–770
    - matching, 472
    - raw data, 780
    - retrieval, 766, 770–776
    - storage in Generic Conduit Base Classes environment, 766, 770, 776–779
    - storage in MFC base class environment, 790, 791, 792–796
    - tab-delimited record, 770, 773, 775–776
    - unchanged record, 765
  - slow, 728, 881
  - speed, 728, 729, 730, 867, 881
  - starting, 727
  - system preference, 495
  - transaction-based, 729, 884
  - user validation, 728
- SynchronizeAppInfo **method, 789**
- Syncmgr.h **file, 753, 755, 760, 800**
- SyncOpenDB **function, 804–805**
- SyncPurgeAllRecs **function, 811**
- SyncPurgeAllRecsInCategory **function, 811**
- SyncPurgeDeletedRecs **function, 811**
- SyncReadNextModifiedRec **function, 807–808, 810**
- SyncReadNextModifiedRecInCategory **function, 804, 807–808**
- SyncReadNextRecInCategory **function, 804, 807**
- SyncReadRecordById **function, 810**
- SyncReadRecordByIndex **function, 810**
- SyncRegisterConduit **function, 804**
- SyncResetRecordIndex **function, 807**
- SyncWriteRec **function, 810–811**
- SyncYieldCycles **function, 811–812**
- SysAlarmTriggeredParamType **structure, 364**
- SysAppLaunch **function, 80, 299–301**
- sysAppLaunchCmdAlarmTriggered **routine, 360, 362, 364**
- sysAppLaunchCmdAskUser **routine, 649–650**
- sysAppLaunchCmdAttention **routine, 376, 379, 380**
- sysAppLaunchCmdCardLaunch **routine, 571**
- sysAppLaunchCmdDisplayAlarm **routine, 360, 362, 364, 366**
- sysAppLaunchCmdFind **routine, 556, 557–561**
- sysAppLaunchCmdGoto **routine, 556, 561–564, 666, 831, 832**
- sysAppLaunchCmdLookup **routine, 299**
- sysAppLaunchCmdNormalLaunch **routine, 300, 571, 706, 831, 832**
- sysAppLaunchCmdNotify **routine, 575, 576**
- sysAppLaunchCmdReceiveData **routine, 650, 655**
- sysAppLaunchCmdSaveData **routine, 556–557**
- sysAppLaunchCmdSyncNotify **routine, 574, 623**
- sysAppLaunchCmdSystemReset **routine, 133, 146, 366**
- sysAppLaunchCmdTimeChange **routine, 366**
- SysAppLauncherDialog **function, 299**
- SysBatteryInfo **function, 304**
- SysBatteryInfoV20 **function, 305**
- SysBroadcastActionCode **function, 302**
- SysCurAppDatabase **function, 485, 824**
- SysDisplayAlarmParamType **structure, 364**
- SysGetModuleDatabase **function, 81, 200**
- SysGetRefNum **function, 81, 200**
- SysGetROMToken **function, 306–307**
- sysGetROMVerBuild **macro, 269**
- sysGetROMVerFix **macro, 269**
- sysGetROMVerMajor **macro, 269**
- sysGetROMVerMinor **macro, 269**

sysGetROMVerStage **macro, 269**  
SysHandleEvent **function, 83, 84, 294**  
SysLibFind **function, 707**  
sysNotify **notifications, 573**  
SysNotifyRegister **function, 573–574, 575**  
sysNotifyVolumeMountedEvent **object, 575**  
sysNotifyVolumeUnmountedEvent **object, 575**  
SysRandom **function, 302–303**  
sys\_socket.h **file, 706, 722, 723**  
SysStringByIndex **function, 159**  
**system preference, 495, 496–497, 548**  
**system resource, 882**  
SystemMgr.h **file, 268, 269, 271, 305**  
SystemPreferencesChoice **enumeration, 496**  
SystemPreferencesType **structure, 496, 497**  
SystemResources.h **file, 678, 852**  
SysTicksPerSecond **macro, 368**  
SysTime **macros, 83, 369**  
SysUIAppSwitch **function, 299, 300–301**

## T

### table

- adding, 489, 490–491
- cell
  - drawing, 390, 440
  - value, setting, 398, 421
- check box, inserting, 389, 390, 399
- class, 883
- color table, 323–325, 335, 863
- column
  - adding, 491–492
  - data size maximum, setting, 491
  - drawing, 440
  - ID, 491, 492, 552
  - information about, returning, 492
  - load routine, custom, 864
  - pop-up trigger, inserting, 399
  - record title, loading to, 439
  - removing, 492
  - spacing, 401, 424, 425
  - text, loading with, 401–402
  - usability, 400–401, 411, 422
  - value, returning, 552
  - value, setting, 398
  - variable, binding, 552–553
  - visibility, 410–414
  - width, 423–425, 441
- conduit, table-based, 744
- date, inserting, 389, 390–391, 398–399
- defined, 883
- deleting, 492
- drawing
  - cell, 390, 440
  - column, 440
  - item, 389, 391, 406–408
  - redrawing, 410, 413, 459, 463
  - row, 405, 410, 432, 438–440
- edit mode, 463, 866
- enumerating, 492
- event handling, 211, 212, 213, 409–410, 463–464
- feature table, 268, 867
- field
  - focus, 463, 467
  - handle, 393–394, 463
  - label, 391
  - note, 399
  - referencing, 461
  - spacing, 401, 452
  - uniqueness, 461
- icon, inserting, 391, 392, 399, 406, 411
- initializing, 190, 392–393, 421–425, 441–442
- item
  - : (colon) display, suppressing, 392, 421
  - drawing, 389, 391, 406–408
  - integer value, associating with, 389, 391
  - introduced, 389
  - note, inserting, 389, 391
  - pop-up list, calling from, 389, 391
  - selection, 880
  - style, 389–390, 398
  - time, inserting, 389, 391
  - user interface element contained, defining, 389
- manager source code, 387
- naming, 491
- pointer, 389–390, 391
- pop-up
  - list, calling from table item, 389, 391
  - trigger, inserting in column, 399
- property, assigning/returning, 389
- record
  - column, loading record title to, 439
  - row, relation to, 415
  - synchronization, storage in table during, 790, 791, 792–796
  - updating, 554
- row
  - adding, 491, 553–554
  - deleting, 453, 554
  - drawing, 405, 410, 432, 438–440
  - field spanning multiple rows, 450
  - height, 400, 429, 432, 442, 450–451

- hiding empty, 432, 452
- ID, 432, 438, 458
- iteration, 421–422
- number displayed simultaneously, returning
  - maximum, 428–429
- record, relation to, 415
- removing, 554
- text, loading with, 401–402
- usability, 401, 422, 432, 450, 458
- value, returning, 432, 552
- value, setting, 398
- visibility, 401, 410–414, 432, 452
- saving, 403–405, 461, 463, 864
- schema, 490–491, 790, 792–793, 795–796
- scrolling, 20, 440–441, 454–455, 458–461
- searching by category offset, 430
- sizing, 189
- text
  - edit indicator, 467–468
  - font, 389, 391, 421–422
  - handle, 393–394, 463
  - item, 391–392, 398, 399
  - label, 389, 391, 393
  - loading text field, 401–403
  - note icon, 391
  - saving, 403–405, 461
  - scrolling text field, 440–441, 458–459
- TableDrawItemFuncType function, 406**
- TableLoadDataFuncType function, 401, 402**
- TableSaveDataFuncType function, 404, 461**
- tables.h file, 798**
- tallCustomTableItem style, 390**
- tap, stylus, 883**
- target field, 619, 621, 622, 636**
- Target Settings dialog box (CodeWarrior), 58–59**
- tblEnterEvent object, 211, 213, 409–410**
- tblExitEvent object, 212, 213**
- TblFindRowData function, 432**
- TblFindRowID function, 432**
- TblGetColumnWidth function, 425**
- TblGetCurrentField function, 405, 461**
- TblGetItemFont function, 389**
- TblGetItemInt function, 389**
- TblGetItemPtr function, 389**
- TblGetLastUsableRow function, 458**
- TblGetRowData function, 432, 438**
- TblGetRowHeight function, 450**
- TblGetRowID function, 432, 438, 458**
- TblGrabFocus function, 463**
- TblInsertRow function, 452**
- TblMarkRowInvalid function, 405**
- TblRedrawTable function, 410, 459**
- TblReleaseFocus function, 463**
- TblRemoveRow function, 453**
- TblRowUsable function, 450**
- tblSelectEvent object, 211, 213, 409**
- TblSetColumnEditIndicator function, 467**
- TblSetColumnSpacing function, 401, 424**
- TblSetColumnUsable function, 400, 411**
- TblSetColumnWidth function, 424**
- TblSetCustomDrawProcedure function, 406, 425**
- TblSetItemFont function, 389, 422**
- TblSetItemInt function, 389, 398, 410**
- TblSetItemPtr function, 389**
- TblSetItemStyle function, 389, 398**
- TblSetLoadDataProcedure function, 398**
- TblSetLoadProcedure function, 401**
- TblSetRowData function, 432**
- TblSetRowHeight function, 432**
- TblSetRowStaticHeight function, 400**
- TblSetRowUsable function, 400, 432**
- TblSetSaveDataProcedure function, 398, 403–404, 461**
- TCP (Transmission Control Protocol), 704**
- TCP/IP (Transmission Control Protocol/Internet Protocol)**
  - bandwidth considerations, 704
  - byte order, 712
  - memory usage, TCP/IP stack, 257
  - packet receipt acknowledgement, 704
  - POSE network library redirection to, 124, 130
  - socket, 715
  - support, 37–38
  - transmission verification, 704
  - UDP compared, 704
- TD (transmit data) signal, 673**
- telephony manager, 38**
- text**
  - anti-aliasing, 345
  - autoshifting, 467, 860
  - carriage return, adding/removing during synchronization, 780
  - case, converting, 845
  - character
    - digit, testing if character is, 286, 844
    - encoding, 843, 845–846, 849–850, 862, 873
    - macro, 285–288
    - size in bytes, returning, 848
  - text buffer, returning character from, 847–849
  - text buffer, transliterating character in, 845

## text (continued)

---

### text (continued)

- color, 226, 327–328, 336
- comparing, 285, 508, 513, 844
- copying, 308–309
- drawing, 274, 275, 279, 332–333, 347
- field
  - capitalization, automatic, 183
  - color, 336
  - copying, 308
  - deleting, 229
  - edit mode, 463, 866
  - focus, 868
  - font, 391, 836
  - form, adding to dynamic, 841
  - handle, 227, 229, 393–394, 405, 463
  - label, 391
  - modifying, 228–229
  - offset, 533
  - returning, 97, 229–230
  - underlining, 183, 336
- font
  - adding custom in Garnet environment, 835–836
  - alternate, 281
  - application preference, storing, 493
  - ascender, 282, 837
  - baseline, 274, 860
  - bitmap, 273–275, 282, 838
  - bold, 280, 422
  - bounding box, returning, 282
  - button, 218
  - color, 226, 327–328, 336
  - control, 836
  - creating custom, 836–839
  - CSS, formatting using, 280, 284
  - current, setting/returning, 274, 282
  - definition file, 837
  - descender, 275, 282, 837, 864
  - determining available, 283–284
  - drawing, 274, 275, 279, 332–333, 347
  - face, 280
  - family, 280–281, 282, 283
  - field, text, 391, 836
  - fitting within specified width, 276–279, 283
  - form, 180
  - handle, 281–282, 283
  - hinting, 345
  - ID, 274
  - italic, 280
  - Librarian application, 194, 421–422, 493
  - list, 836
  - memory, releasing, 282
  - oblique, 280
  - resource, 835
  - scalable, 274, 279–284
  - size, returning, 274–275, 282, 424
  - size, setting, 280–281, 282, 837
  - style, 280, 282
  - table, 389, 391, 421–422
  - TrueType, 274, 281, 835
- hinting, 345
- integer value, converting string to, 285
- `KeyDownEvent` object, testing incoming character
  - from, 287
- label
  - button, 31, 182, 186, 188, 218
  - check box, 23
  - defined, 871
  - dialog box, 36
  - field, text, 391
  - form, 184–185, 841
  - justification, 36
  - pop-up trigger, 186, 246
  - selector trigger, 24, 188–189, 220–221
  - table, 389, 391, 393
  - volume, 579–580, 584, 595
- list
  - passing string array to, 243
  - string list, 159–160, 859, 882
- manager, 843–844
- memory
  - handle, text field, 227, 229, 393–394, 405, 463
  - iterating through text buffer, 847–848
  - returning character from text buffer, 847–849
  - transliterating character in text buffer, 845
  - word bounds, returning, 848–849
- record, editing string in-place, 528–531, 871
- resource, string, 158–160, 166
- screen width, fitting to, 276–279, 283
- searching, 557, 561, 844–845
- sorting, 285
- table
  - edit indicator, 467–468
  - font, 389, 391, 421–422
  - handle, 393–394, 463
  - item, 391–392, 398, 399
  - label, 389, 391, 393
  - loading text field, 401–403
  - note icon, 391
  - saving, 403–405, 461
  - scrolling text field, 440–441, 458–459

transliterating, 845–847  
 underlining, 183, 332–333, 336  
**TextMgr.h file**, 286, 639, 845, 849  
**textTableWidgetItem style**, 389, 391, 398  
**textWithNoteTableWidgetItem style**, 389, 391, 398, 463  
**32-bit IP address**, 709, 711  
**32K jump limit**, 827, 828–830  
**threading**, 15, 86  
**tick (time value)**, 83, 303, 367, 368, 883  
**TicTacToeDraw function**, 234–236  
**TicTacToeTap function**, 237–239  
**TimAdjust function**, 371  
**TimDateTimeToSeconds function**, 369  
**time**  
   calculation involving, 368–371  
   clock, 367  
   Cobalt time macros, 368–369  
   database timestamp, 483  
   DST, 372–374  
   formatting, 223, 371–374  
   picker, 221, 223–225, 883  
   returning, 368  
   separator character, 373  
   setting, 368  
   table item, inserting in, 389, 391  
   tick, 83, 303, 367, 368, 883  
   user preference, 371–374  
   zone, 372, 373–374  
**time field**, 619  
**TimeFormatType enumeration**, 373  
**timeTableWidgetItem style**, 389, 391  
**TimeType structure**, 223–224, 367–368  
**TimGetSeconds function**, 223, 368  
**TimGetTicks function**, 303, 368  
**TimSecondsToDateTime function**, 223, 369  
**TimSetSeconds function**, 368, 373  
**title bar**  
   alert, 158, 166  
   form, 18, 31, 840, 883  
   screen, 34  
**To Do application**, 247  
**ToggleColumn function**, 411  
**ToggleColumns function**, 413–414  
**ToggleRow functions**, 411–413  
**tracing**, 883  
**track (Multimedia Services)**, 353, 355  
**transfer mode**, 326–327, 884  
**transferring data**. *See* exchanging data  
**transformation, drawing**, 345  
**Transmission Control Protocol (TCP)**, 704

**Transmission Control Protocol/Internet Protocol.**

**See TCP/IP**

**transmit data (TD) signal**, 673

**Treo handheld**, 813

**trigger**

  pop-up

    anchoring, 186

    category, 186, 547

    defined, 876

    form, adding to dynamic, 841

    icon, 21

    label, 186, 246

    list, 21, 185, 186, 245–246

    placement, 22

    selector, 23, 24, 188–189, 220–225, 841

**TrueType font**, 274, 281, 835

**Tungsten handheld**, 267, 270, 813

**TxtCaselessCompare function**, 844

**TxtChar macros**, 286–287

**TxtCharEncoding function**, 849

**TxtCharIsDigit macro**, 844

**TxtCompare function**, 844

**TxtFindString function**, 844–845

**TxtGetChar function**, 847

**TxtGetNextChar function**, 847–848

**TxtGetPreviousChar function**, 847–848

**TxtGetTruncationOffset function**, 848

**TxtGlue functions**, 850

**TxtGlueCharIs macros**, 286–287

**TxtGlueGetHorizEllipsisChar function**, 279

**TxtGlue.h file**, 286

**TxtNextCharSize macro**, 848

**TxtPreviousCharSize macro**, 848

**TxtStrEncoding function**, 849

**TxtTransliterate function**, 845–847

**TxtWordBounds function**, 848–849

**type field**, 471, 473, 620, 636

**typed data object**, 617, 884

## U

**UART (Universal Asynchronous Receiver and Transmitter) chip**, 673, 674, 884

**UDP (User Datagram Protocol)**, 704, 715

**UI Process**, 15

**UIAS (User Interface Application Shell)**, 884

**UIColorGetTableEntryIndex function**, 337, 339

**UIColorGetTableEntryRGB function**, 337, 339

**UIColorSetTableEntry function**, 339

**UIPickColor function**, 225

**Ultra data exchange mode, 619**

**underscore, double ( `__` ) system function prefix, 832**

**Uniform Resource Locator (URL), exchange library, 637–640**

uniqueID field, 472, 663

uniqueIDSeed field, 471

**Universal Asynchronous Receiver and Transmitter (UART) chip, 673, 674, 884**

**universal connector, 567. See also serial communication**

**UNIX Berkeley sockets compatibility, 722**

UnpackRecord function, 504–506

UpdateTables function, 750

**URL (Uniform Resource Locator), exchange library, 637–640**

**USB (Universal Serial Bus)**

baud rate, 679

cradle, handheld, 673

opening connection, 679

startup task, 680

**User Datagram Protocol (UDP), 704, 715**

**user interface. See also specific interface element**

color, 335–339

dynamic, 839–843, 866

ease of use, 31–32

guidelines, 28–37, 151–152

highlighting frequently used feature, 30

speed, 29–30

table item, defining user interface element contained by, 389

**User Interface Application Shell (UIAS), 884**

**user preference, 371–374**

## V

**variable**

column, binding, 552–553

gadget state, storing in, 233

Garnet environment, considerations when using global variable in, 576

memory, allocating, 233

network library global variable requirement, 706

vchrJog events, 823

vchrJogBack event, 819

vchrLowBattery event, 306

**version**

application requirement, checking at launch time, 301–302

CodeWarrior

full version, 725

Mac OS version, 117

conduit version, returning, 750, 754

database, 482, 531–536

determining, 268–271

exchanging data, checking when, 654

feature, 268

graphic backward compatibility, 311–312

header file, version 3.5 differences, 111–113

MFC version, returning, 753

overview of Palm OS versions, 39–43

resource

assigning resource version string, 154

targeting Palm OS version, 155–156

serial manager, 676–677, 698–701

system version number, 270–271

VFS manager, 572

version field, 471

**VFAT (Virtual File Allocation Table), 568, 584**

**VFS Test application**

alert, 585

directory creation, 581–584

error handling, 585–586

feature overview, 571–572

file

reading, 599–603

writing, 607–610

PilotMain function, 575

StartApplication function, 574–575

volume

name display, 576

referencing, 578, 579

**VFS (Virtual File System) manager**

directory, current, 581

error handling, 585–586

existence, checking, 572

file system support, 569

notification, 573, 574–575

path name, 581

secondary storage card interface, as, 568

version, 572

volume

information, returning, 580–581

referencing, 576–579

**VFSDirCreate function, 581, 584**

**VFSDirEntryEnumerate function, 610–611**

**VFSExportDatabaseToFile function, 615**

**VFSExportDatabaseToFileCustom function, 615**

**VFSFileClose function, 589, 594**

**VFSFileDBGetRecord function, 614, 615**

**VFSFileDBGetResource function, 615**

**VFSFileDBInfo function, 614–615**

**VFSFileDelete function, 589, 593**

**VFSFileEOF function, 605**  
**VFSFileGetAttributes routine, 595**  
**VFSFileGetDate function, 596–597**  
**VFSFileOpen function, 593–594, 604**  
**VFSFileRead function, 598, 603, 604**  
**VFSFileReadData routine, 598–599, 604**  
**VFSFileSeek function, 604–605**  
**VFSFileSetAttributes routine, 596**  
**VFSFileSetDate function, 597**  
**VFSFileSize function, 597–598**  
**VFSFileTell function, 604**  
**VFSFileWrite function, 604, 607, 610**  
**VFSGetDefaultDirectory function, 613**  
**VFSImportDatabaseFromFileCustom function, 615**  
**VFSMgr.h file, 580**  
**VFSVolumeEnumerate function, 576–577**  
**VFSVolumeGetLabel function, 579**  
**VFSVolumeInfo function, 580**  
**VFSVolumeSetLabel function, 579, 580**  
**vibrating handheld, 377**  
**video, 353**  
**View ⇄ Databases (Palm OS Simulator), 137**  
**View ⇄ Events (Palm OS Simulator), 139**  
**View ⇄ Heaps (Palm OS Simulator), 138**  
**View ⇄ Lock (Palm OS Simulator), 138**  
**Virtual File Allocation Table (VFAT), 568, 584**  
**Virtual File System manager. See VFS manager**  
**VisualCafé development environment, 725**  
**volume**  
    audio, 24, 352  
    storage unit  
        defined, 884  
        directory, creating on, 581–584  
        event handling, 573, 575  
        expansion card, 568  
        finding all mounted volumes, 577  
        iteration, 577  
        label, 579–580, 584, 595  
        media type, returning, 580  
        mounting/unmounting, 568, 573, 574, 575, 576  
        name, 576, 579–580  
        pointer, 577  
        read-only, 569  
        referencing, 576–579, 581  
        size, returning, 580  
        space available, returning, 580  
        writable, 569  
**VolumeInfoTag structure, 580–581**  
**VolumeInfoType structure, 580–581**  
**volume.nam function, 580**

## W

**warning dialog box alert, 19**  
**.wav files, 163, 352**  
**WebGain VisualCafé development environment, 725**  
**WHERE SQL statement, 551**  
**while loop, 432, 450, 511**  
**WinCopyRectangle function, 317**  
**WinCreateOffscreenWindow function, 316**  
**WinCreateWindow function, 313, 314, 315**  
**WinDeleteWindow function, 314, 315**  
**window**  
    active, 312–313, 859  
    bounds, returning, 317  
    closing, 314, 315  
    color, 317, 318  
    creating, 313, 314–315  
    defined, 884  
    drawing, 312–313, 315, 318–319, 866  
    frame, 313–314, 315, 868  
    introduced, 88  
    memory  
        buffering, double, 316, 865  
        removing window from, 314  
    offscreen, 314, 316–318, 874  
    pop-up, 314–316  
**Window ⇄ Catalog (Constructor), 171**  
**Window ⇄ Reset Perspective (Eclipse Workbench), 72**  
**Window ⇄ Show View (PODS), 144**  
**Window.h file, 313**  
**Windows Registry conduit entry. See registering conduit**  
**WinDrawBitmap function, 333**  
**WinDrawChar function, 333**  
**WinDrawChars function, 185, 236, 245, 275, 332**  
**WinDrawGrayLine function, 328**  
**WinDrawGrayRectangleFrame function, 329, 332**  
**WinDrawLine function, 328–329**  
**WinDrawPixel function, 328**  
**WinDrawRectangle function, 329**  
**WinDrawRectangleFrame function, 236, 332**  
**WinDrawTruncChars function, 275–276**  
**WinEraseChar functions, 333**  
**WinEraseLine function, 328, 329**  
**WinErasePixel function, 328**  
**WinEraseRectangle function, 236, 329**  
**WinEraseRectangleFrame function, 332**  
**WinFillLine function, 329**  
**WinFillRectangle function, 330**  
**WinGetBounds function, 317**  
**WinGetDrawWindowBounds function, 317**

## WinGetPattern function

---

WinGetPattern **function, 331**  
WinGetSupportedDensity **routine, 341**  
WinGetWindowFrameRect **function, 314, 315**  
WinIndexToRGB **function, 325–326**  
WinInvertChar **functions, 333**  
WinInvertLine **function, 329**  
WinInvertPixel **function, 328**  
WinInvertRectangle **function, 329**  
WinPaintBitmap **function, 327**  
WinPaintChar **function, 333**  
WinPaintLine **function, 329**  
WinPaintPixel **function, 328**  
WinPaintRectangle **function, 329–330**  
WinPalette **function, 335**  
WinPopDrawState **function, 332, 342**  
WinPushDrawState **function, 319, 327, 332, 342**  
WinRestoreBits **function, 315–316**  
WinRGBToIndex **function, 325–326**  
WinSaveBits **function, 314**  
WinScalePoint **function, 343**  
WinScaleRectangle **function, 343**  
WinScreenMode **function, 320–321**  
WinSetActiveWindow **function, 313**  
WinSetBackColorRGB **function, 327**

WinSetCoordinateSystem **function, 342**  
WinSetDrawMode **function, 317, 326–327**  
WinSetDrawWindow **function, 313, 315**  
WinSetForeColor **function, 327, 329**  
WinSetForeColorRGB **function, 327, 329**  
WinSetPattern **function, 329, 330–331**  
WinSetTextColorRGB **function, 327**  
WinSetUnderlineMode **function, 332–333**  
**Wireless Internet Feature Set, 884**  
WriteDatabase **function, 669–670**  
WriteFile **function, 607–610**  
WriteOutData **method, 777, 779**  
WriteRecord **routine, 778–779**  
WriteSerial **function, 683–684**

## X

xprf **resource, 156**  
XRD (XML Resource Description) files, **62, 163–164, 176**

## Z

Zire handheld, **267, 270, 271, 813**