

12 Anhang Programmier-Tipps

Symbols

#MODIFIED = Feldänderung 220
\$attributes, Objektattribute 341
\$bringtofront - Fenster heranholen 320
\$cinst und \$cinst() 59, 148
\$ctarget = aktives Feld im Fenster 241
\$ctarget, \$hasfocus 349
\$filelist 41
\$filter 103
\$hasfocus, \$ctarget 349
\$open mit Parametern 497
\$openonce 386
\$redraw und \$bobjs 281
\$ref 50
\$selectdistinct 412
\$sendall 122, 576
\$smartlist 103, 105, 148, 216
\$smartlist für Cancel 224
\$smartlist, \$filter 178
\$splitpathname 457
\$version als Speicherelement 173

A

aktives Feld im Fenster setzen = \$ctarget 241
Aktivierungsreihenfolge der Fensterobjekte 205
Aktualisierung der Fensteranzeige für Hintergrundobjekte 281
Ändern mehrerer Objekte 166
auch Inoffizielles ausprobieren 99

B

Befehlseingabe 32
Bilder aus der Zwischenablage 193
Breakpoint
 als Anspringstelle für schwer erreichbare Methodendarstellungen 83
 Breakpoint und Debugger während der Programmierung nutzen 92
Breakpoints
 als Anspringstellen 191
 als Eselsbrücken 187, 277, 410, 416

C

Cascade-Property bei Menüzeilen 98
chr(13) = CR stellt Zeilenvorschub dar 174
chr(39) stellt Hochkomma dar 80, 174
Code-Class 30
Combos und \$obj 383
Computer simulieren lassen 126

D

Debugger als aktives Instrument 314
Design-Modus und Fenstermaximierung 350
Dokumentation aktuell halten 118
Drag&Drop 413

E

Editoren-Programmierung 439
Eigenschaften von Fensterobjekten per Notation verändern 202
Endlosschleifen beenden mit der Pause-Taste 531, 563
Events aussuchen über Ausprobieren 195, 341, 389, 420, 497, 543
External commands 143
externe Komponenten = Menüaufruf im Browser 146
externe Komponenten = müssen geladen werden 146
externe Komponenten = Windows-Elemente im Component-Store 146

F

Fehlersuche 353
Feldliste zum Aktivieren von Fenster-Feldobjekten 192
Feldobjektliste nach Funktionsschlüsseln 191
Feldreferenzen müssen Feldcharakteristik haben 278
Fenster mit Tab-Buttons 159
field reference 50, 88
FileOps-betriebssystemunabhängige Dateizugriffe 145
Find&Replace vor Namensänderungen 302
fixe Breakpoints bei unfertigen Programmteilen 77, 78
FondOps erlaubt flexible Textbehandlung 183

G

Graphik
Daten 337
Farbverteilung 339
Koordinaten-Ausmaße 339
Koordinatentext-Ausmaße 339
selectobjects selektiert Details 336
Untertitel-Ausmaße 339
Graphik-Objekte 351, 358, 362
Graphk
Koordinatentext 340

H

hängende Toolbar 381
Hash-Variablen vermeiden 54

I

Icon-Auswahl 168
Indirektion mit eckigen Klammern=Brackets 55, 78
Indirektionen auch um Methoden 455
Instanzen kontrollieren 268, 269
interne Toolbar, Instanzierung 384
item reference 50

J

jst() zur Textaufbereitung 307

K

kBMcolorpicker = Farbauswahl 344
keine Objektnamen mit führenden Zahlen 78
Key Events aktivieren 543
Klassenmethoden von außen ansprechen 172
Kommentare sorgfältig setzen 75
Konstante als Einzelfall betrachten 85
Kopierbarkeit von Buttons 189
Kopieren bestehender Anwendungen 6, 7
kopierte Instanzvariable in vererbten Objekten prüfen 66

L

leere Listen direkt befüllen 239
Listen als flexible Textgestalter 201
Listenverhalten beeinflussen 195
Listenzugriffe direkt 54, 88
Lock von Objekten vermeiden 389

M

mehrgleisige Methoden 418
Menü als Popup 344
Menü komponieren 347
Methoden wechseln zwischen Klasse und Feldern 160
Methodenobjekte beim Erstellen des Datensatzes berücksichtigen 474
Methodentext beschaffen 174
Methodentext für einen eigenen Kommentarzeile komponieren 174
Methodentext komponieren 328

mouseover nie mit Breakpoints prüfen 421
 mouseover zur Positionsermittlung 421

N

Notation bevorzugen 65
 Notation unter Verwendung eigener \$-Namen mit // 174
 NULL einzeln abfragen 155, 254, 257, 269, 284

O

Objektdetails in Übersichten helfen 390
 Objekte in rekursivem Programmablauf 124

P

Platzhalter = Wildcards weiten Suchaktionen aus 195
 Problematische Syntax bereinigt ,//^ 47
 Programm-Code lesbar gestalten 129

R

Raster auf Windows 165
 Rechte Maus und Dateneingaben entgegennehmen 221
 Rückspeicherung der Datendatei aktivieren 499, 523
 Runtime contra Design 389

S

Schnell-Kopie 168
 Sortieren von Listen 196
 Spaltenüberschriften bei Listen 191
 SQL Cursor 80
 SQL-Command zusammensetzen 223
 SQL-Delete nie ohne vorherigen Select 476
 SQL-Probleme bei Differenzen Schema-Table 313
 SQL-Probleme bei Überdefinition der SQL-Liste 277
 Statusleisten benutzen 164
 Stopp eines unerwünschten Prozesses mit der Pause-Taste 38, 563
 style() zur Textaufbereitung mit Icons, Mindestlänge 376
 style() zur Textaufbereitung, auch mit Icons 317, 318, 367
 Subwindows = Fenster im Fenster 160
 sys(94) = Separator 145

T

taskeigene Menüs und Toolbars 406
 Test mit Programmierung koppeln 73, 120
 Test mithilfe von Hash-Variablen 55, 92, 97, 119
 Testcode 30
 Texte separieren 376
 Tooltip = der kleine Hilfetext vor Ort 161
 Tooltip-Texte 34

U

Überflüssige Variable entfernen = delete unused variables 77
 Überschreiben von Omnis-eigenen Befehlen 279
 Unberührbare bei Source-Manipulation 381, 398
 Universalgenie Binary-Feld 83, 137, 143, 208

V

Variable als Verarbeitungspuffer 204
 Variablenübersicht und Variablen-Initialisierung 93
 variables Popup-Menü 98, 99, 100
 Verarbeitung über Stack beenden 53
 Verhalten von Fensterobjekten bei Größenänderung 165
 Vorab-Test mit Debugger und Rechter Maus 112
 Vorgehen bei Testdaten der 4ff-Applikation 518
 Vorsich
 mit \$fetch, der Option bAppend und einem darauffolgenden \$dowork 554
 Vorsicht
 mit \$fetch, nicht zuviele Datensätze unkontrolliert einlesen 554

mit doppelten Popup menu from list 98
mit Klassen-Manipulationen 394

W

Wechsel der Anzeigarten 360

Window öffnen, schließen, minimieren und vergrößern lassen 164

Workaround 285