

Actuate



*Developing Customized Reports
using Actuate e.Report Designer Professional*

Information in this document is subject to change without notice. Examples provided are fictitious. No part of this document may be reproduced or transmitted in any form, or by any means, electronic or mechanical, for any purpose, in whole or in part, without the express written permission of Actuate Corporation.

© 1995 - 2004 by Actuate Corporation. All rights reserved.
Printed in the United States of America.

Contains information proprietary to:

Actuate Corporation
701 Gateway Boulevard
South San Francisco, CA 94080
<http://www.actuate.com>

The software described in this manual is provided by Actuate Corporation under an Actuate License agreement. The software may be used only in accordance with the terms of the agreement.

Actuate Corporation trademarks and registered trademarks:

Actuate, the Actuate logo, e.Analysis, Correlink, Correlink Framework, e.Report, e.Reporting, e.Spreadsheet, Encyclopedia Formula One, Internet Spreadsheet, Lens, Lens Builder, Live Report Document, Nimble, the Nimble logo, Nimble Technology, Pace, ReportCast, Report Encyclopedia, ReportingEngines, the ReportingEngines logo, Reportlet, Spreadsheets Everywhere, Tidestone Technologies, and XML reports.

Third party trademarks or registered trademarks of their respective owners, companies, or organizations:

Apache Software Foundation (<http://www.apache.org/>): Axis, Crimson, Jakarta-ORO, Tomcat, Xalan, Xerces, and XML4C. Apple Computer, Inc.: Mac OS, Macintosh, TrueType. BEA Systems, Inc.: BEA Liquid Data for WebLogic, BEA WebLogic Portal, BEA WebLogic Portal Server, BEA WebLogic Server, BEA WebLogic Workshop. Bits Per Second, Ltd. and Graphics Server Technologies, L.P.: Graphics Server. Borland Software Corporation: JBuilder. Bruno Lowagie and Paulo Soares: iText, licensed under the Mozilla Public License (MPL). Codejock Software: Xtreme Toolkit. Component One, LLC.: VSFlexGrid Pro. DataDirect Technologies Corp: DataDirect Connect. Fred L. Drake, Jr. (<http://sourceforge.net/projects/expat/>): Expat XML parser, created by James Clark, licensed under the MIT License. Hewlett-Packard Company: HP-UX. IBM Corporation: 1-2-3, AIX, DB2, Informix-ESQL/C, ICU, Lotus, and WebSphere. Indiana University Extreme! Lab (<http://www.extreme.indiana.edu>): XML Pull Parser and XPP. InstallShield Corporation: InstallShield. InterNetivity Inc.: Databeacon. JDBM Project (<http://jdbm.sourceforge.net>): JDBM. LEAD Technologies, Inc.: LEADTOOLS. Linus Torvalds: Linux. Lucent Technologies, Inc.: FPDTOA, created by David M. Gay. Microsoft Corporation: ActiveX, Microsoft, MS-DOS, MSN, Windows. Nanode LLC: OpenJNLP, licensed under the Mozilla Public License (MPL). Netscape Communications Corporation, Inc.: Netscape, Netscape Communications, Netscape Communicator, Netscape Enterprise Server, and Netscape Navigator. The OpenSSL Project (<http://www.openssl.org/>): Mac OS X, Mac OS X Server, OpenSSL Toolkit. Oracle Corporation: Oracle Call Interface. Progress Software Corporation: Progress. Quadralay Corporation: WebWorks. Rogue Wave Software, Inc.: NobleNet RPC and Rogue Wave SourcePro. SAP AG: SAP, SAP BW, SAP R/3. Sun Microsystems, Inc.: 100% Pure Java, iPlanet, J2EE, Java and all Java-based marks, JavaServer Pages, ONC, Solaris, SPARC, Sun, Sun Microsystems, and Sun ONE. Sybase, Inc.: CT-Library. Symantec Corporation: Visual Cafe. Unicode, Inc.: Unicode. World Wide Web Consortium (W3C): HTML Tidy and tidy.c. X/Open Company, Ltd.: UNIX. Zero G Software, Inc.: InstallAnywhere. Zope Corporation: DCLC and Digital Creations.

All other brand or product names are trademarks or registered trademarks of their respective owners, companies or organizations.

Document No. 050111-5-130322 February14, 2005

Contents

| | |
|--|-----------|
| Chapter 1 | |
| Setup instructions | 1 |
| Hardware requirements | 2 |
| Software requirements | 2 |
| Course materials for students | 2 |
| Setting up student workstations | 3 |
| Setting options in e.Report Designer Professional | 3 |
| Configuring an ODBC data source | 11 |
| Verifying access to the Sfdata sample database | 13 |
| Setting a display appearance option in Microsoft Windows | 13 |
| Creating a user and a home folder using Actuate Management Console | 14 |
| Chapter 2 | |
| Course introduction | 19 |
| Developing Customized Reports | 20 |
| About this course | 20 |
| Prerequisites | 20 |
| Course objectives | 22 |
| What you learn | 22 |
| Course agenda – day 1 | 23 |
| Course agenda – day 2 | 24 |
| Actuate course curriculum | 25 |
| Working with Actuate e.Report Designer Professional | 25 |
| Developing Customized Reports | 26 |
| Designing and Publishing e.Spreadsheet Reports | 26 |
| Developing Customized Spreadsheet Reports | 26 |
| Working with Formula One e.Report Designer | 26 |
| Actuate course curriculum (continued) | 27 |
| Managing Actuate iServer System | 27 |
| Creating an e.Reporting Web Site using Actuate ReportCast | 27 |
| Customizing Actuate Active Portal for JSP | 27 |
| Customizing Actuate Active Portal for .NET | 28 |
| Actuate Information Delivery API Workshop | 28 |
| Integrating and Deploying Formula One e.Report Engine | 28 |
| Supporting information | 29 |
| Chapter 3 | |
| Understanding the Actuate framework | 31 |
| Understanding the Actuate framework | 32 |
| Overview | 33 |
| About the Actuate framework | 33 |
| What you learn | 33 |
| Key term | 34 |
| Supporting information | 34 |
| Using the framework | 35 |
| Using the framework to design custom reports | 35 |
| About Actuate Basic fundamentals | 35 |
| About design and development tasks | 35 |
| Understanding object-oriented concepts and terminology | 36 |
| About object-oriented concepts and terminology | 36 |

| | |
|--|-----------|
| About class types | 38 |
| About abstract classes | 38 |
| About concrete classes | 38 |
| Categorizing AFCs. | 39 |
| About data stream classes | 39 |
| About section classes | 39 |
| About control classes | 39 |
| About page layout classes | 40 |
| Understanding class protocol. | 41 |
| Relating classes to components. | 42 |
| Understanding how classes relate to components | 42 |
| Getting information about a component's superclass | 42 |
| Understanding how class categories relate to components in the toolbox | 42 |
| About the development environment | 43 |
| Summary | 44 |
| | |
| Chapter 4 | |
| Understanding data flow during report generation | 45 |
| Understanding data flow during report generation | 46 |
| Overview | 47 |
| About data flow. | 47 |
| What you learn | 47 |
| Key terms | 47 |
| Supporting information | 48 |
| Understanding how the factory builds a report | 49 |
| About report section processing | 49 |
| About report section slots | 50 |
| About report section classes | 50 |
| Understanding report section classes | 51 |
| About AcReportComponent. | 51 |
| About AcReport. | 52 |
| About AcReportSection | 52 |
| Understanding report processing | 53 |
| About the report build process | 53 |
| About Factory methods | 54 |
| Overriding a report section method | 55 |
| Customizing the Start() method | 56 |
| Obtaining a connection. | 56 |
| Obtaining a data stream | 56 |
| Instantiating a new data stream. | 57 |
| Starting a data stream | 57 |
| Finishing a data stream | 58 |
| Customizing the Finish() method | 58 |
| Understanding a connection class | 59 |
| About AcConnection. | 59 |
| About AcDBConnection | 59 |
| Working with a connection | 61 |
| Using a connection | 61 |
| Opening a connection | 62 |
| Disconnecting | 62 |
| Using IsConnected() | 63 |
| Using RaiseError() | 63 |
| Tracking program flow | 64 |
| Using Debug | 64 |

| | |
|---|----|
| Running a Debug session | 64 |
| Controlling program execution | 64 |
| Using a message | 65 |
| Working with MsgBox | 65 |
| Working with ShowFactoryStatus | 65 |
| Exercise 1 | 66 |
| Examining report section processing | 67 |
| Summary | 76 |

Chapter 5

Developing a report with a custom SQL query.77

| | |
|--|-----|
| Developing a report with a custom SQL query. | 78 |
| Overview | 79 |
| About custom SQL queries | 79 |
| What you learn | 79 |
| Key term | 79 |
| Supporting information | 80 |
| Understanding data adapter classes | 81 |
| About AcDataAdapter | 82 |
| About AcDataSource | 82 |
| About AcDatabaseSource | 82 |
| Understanding data rows | 84 |
| About AcDataRow | 84 |
| Working with columns stored in a data row | 85 |
| Understanding database source classes | 86 |
| About AcQuerySource | 87 |
| About AcSqlQuerySource | 87 |
| About AcSqlQuerySource::ObtainSelectStatement() | 87 |
| Understanding SetSortKey() | 89 |
| About AcDataAdapter::AddSortKey() | 89 |
| About AcTextQuerySource | 90 |
| About AcStoredProcedureSource | 90 |
| About other database source classes | 90 |
| Understanding AcDBStatement and AcDBCursor classes | 91 |
| About AcDBStatement | 91 |
| About AcDBCursor | 92 |
| Controlling SQL processing. | 93 |
| Preparing a SQL statement | 93 |
| Allocating and opening a cursor | 94 |
| Using AcDatabaseSource::OpenCursor() | 94 |
| Using AcDBStatement::OpenCursor() | 94 |
| Using AcDBCursor::OpenCursor() | 94 |
| Binding a data column | 95 |
| Processing a data row | 97 |
| Fetching a data row | 97 |
| Instantiating a new data row | 98 |
| Processing the data row | 98 |
| Using AcDatabaseSource::GetDBConnection() | 98 |
| Overriding the data source's Fetch() method | 98 |
| Exercise 2 | 100 |
| Examining data adapter processing | 101 |
| Working with a static SQL query. | 107 |
| About static SQL queries. | 107 |
| Creating a static SQL query | 107 |

| | |
|--|------|
| Working with a dynamic SQL query | .108 |
| About dynamic SQL queries | .108 |
| Creating a dynamic SQL query | .108 |
| Using a static parameter | .109 |
| Using an ad hoc parameter | .109 |
| Exercise 3 | .110 |
| Customizing SQL query processing | .111 |
| Summary | .116 |

Chapter 6

Using a filter to customize a data stream117

| | |
|--|------|
| Using a filter to customize a data stream | .118 |
| Overview | .119 |
| About data filters | .119 |
| What you learn | .119 |
| Supporting information | .119 |
| Understanding data filters | .120 |
| About single-input filters | .121 |
| About multiple-input filters | .121 |
| Understanding data filter classes | .123 |
| About AcDataFilter. | .124 |
| About AcSingleInputFilter. | .124 |
| Fetching a custom set of data rows | .124 |
| Specifying the input adapter | .125 |
| About AcMultipleInputFilter | .125 |
| Understanding buffer classes. | .126 |
| About AcDataAdapter | .126 |
| About AcMemoryBuffer | .127 |
| About AcRecordBuffer | .127 |
| Exercise 4 | .128 |
| Using a memory buffer | .129 |
| Using a data sorter class. | .138 |
| About AcMemoryDataSorter | .138 |
| Overriding Compare() | .139 |
| Exercise 5 | .140 |
| Re-sorting data in a report | .141 |
| Understanding multiple-input filters | .146 |
| Handling multiple data sources. | .146 |
| About AcMultipleInputFilter | .146 |
| Understanding Start() and Finish(). | .147 |
| Implementing filter logic | .147 |
| Exercise 6 | .148 |
| Joining data rows using a multiple-input filter | .149 |
| Setting up the inputs and output for a multiple-input filter | .150 |
| Designing logic to retrieve columns and join data rows | .152 |
| Summary | .156 |

Chapter 7

Customizing a page.157

| | |
|-----------------------------------|------|
| Customizing a page | .158 |
| Overview | .159 |
| About page customization. | .159 |
| What you learn | .159 |

| | |
|--|------------|
| Supporting information | 159 |
| Understanding page creation | 160 |
| Understanding the page and structure hierarchy | 160 |
| About the page framework | 161 |
| Creating a page | 161 |
| Understanding the page list classes | 162 |
| About AcPageList | 162 |
| About AcSimplePageList | 164 |
| About AcLeftRightPageList | 164 |
| About AcTitleBodyPageList | 164 |
| Understanding the page layout classes | 165 |
| About AcBasePage | 166 |
| About AcPage | 166 |
| About AcSubPage | 166 |
| Understanding the flow classes | 167 |
| About AcFlow | 167 |
| About AcLinearFlow | 168 |
| About AcTopDownFlow | 168 |
| Exercise 7 | 169 |
| Designing a custom subpage that uses multiple flows | 170 |
| Summary | 176 |
| | |
| Chapter 8 | |
| Customizing a control | 177 |
| Customizing a control | 178 |
| Overview | 179 |
| About customizing a control | 179 |
| What you learn | 179 |
| Supporting information | 179 |
| Understanding controls | 180 |
| About data controls | 180 |
| About static controls | 180 |
| Understanding how a control affects report generation time | 180 |
| Understanding control classes | 182 |
| About AcReportComponent | 183 |
| About AcVisualComponent | 184 |
| About AcControl | 184 |
| About AcTextualControl | 185 |
| About AcDataControl | 185 |
| Understanding frame classes | 186 |
| About frames | 186 |
| About AcBaseFrame | 187 |
| About AcDataFrame | 187 |
| About AcFrame | 188 |
| Understanding ValueExp | 189 |
| About ValueExp | 189 |
| Using Expression Builder to develop an expression | 190 |
| Using ValueType | 190 |
| Working with a control | 191 |
| Using DataValue variable to access the value of a control | 191 |
| Using GetControlValue() | 192 |
| Using the ObjectVariable property | 192 |
| Using FindContentByClass() | 192 |

| | |
|---|-------------|
| Exercise 8 | .193 |
| Customizing a control | .194 |
| Summary | .201 |
| Chapter 9 | |
| Developing a report with dynamic geometry | .203 |
| Developing a report with dynamic geometry | .204 |
| Overview | .205 |
| About dynamic geometry | .205 |
| What you learn | .205 |
| Supporting information | .205 |
| Understanding dynamic text controls | .207 |
| About dynamic text controls | .207 |
| Understanding supported text formats | .207 |
| Understanding the differences between a dynamic text control and a static text control. | .208 |
| Understanding the Factory report generation process. | .209 |
| About the Factory report generation process | .209 |
| About the report rendering environment | .210 |
| Working with fonts | .211 |
| Understanding how the Factory gets font information. | .211 |
| Embedding font information in a report object executable (.rox) file. | .212 |
| Creating a customized font file | .212 |
| Working with a dynamic text control | .214 |
| Placing a dynamic text control in a report design. | .214 |
| Understanding how a dynamic text control affects other controls in a frame | .215 |
| Understanding how the Factory can resize and reposition a control. | .216 |
| Repositioning a control that appears above or below a dynamic text control | .216 |
| Resizing and repositioning a control that appears to the right or left of a dynamic text control. | .217 |
| Changing the default sizing and positioning behavior of a control. | .218 |
| Affecting other controls by changing a dynamic text control's size | .219 |
| Dynamically reducing the height of a dynamic text control | .219 |
| Moving a control up when a dynamic text control above it shrinks | .219 |
| Setting the maximum height of a dynamic text control | .220 |
| Resizing a control to match the size of a dynamic text control | .220 |
| Changing a control's size and position relative to a frame | .221 |
| Changing a control's height relative to the frame | .221 |
| Changing a control's position relative to the frame. | .222 |
| Using a dynamic text control in different types of frames. | .224 |
| Using a dynamic text control in a frame in a Content, Before, or After slot | .224 |
| Using a dynamic text control in a frame in a PageHeader slot | .225 |
| Using a dynamic text control in a frame in a PageFooter slot | .225 |
| Customizing how a frame splits across pages | .226 |
| Controlling how a frame and its contents span multiple pages | .226 |
| Specifying how a frame and its contents split across pages | .226 |
| Splitting a frame that contains a dynamic text control | .227 |
| Excluding an area of the frame from splitting. | .228 |
| Controlling how the Factory splits a paragraph. | .229 |
| Changing the margins of a dynamic text control. | .230 |
| About dynamic text control margins | .230 |
| Setting margins for the segments of a split dynamic text control | .231 |
| Exercise 9 | .232 |
| Developing a report with dynamic geometry | .233 |
| Controlling page splitting | .238 |
| Summary | .241 |

| | |
|--|------------|
| Chapter 10 | |
| Customizing a subreport | 243 |
| Customizing a subreport | 244 |
| Overview | 245 |
| About subreports | 245 |
| What you learn | 245 |
| Supporting information | 245 |
| Understanding sections | 246 |
| About sections | 246 |
| Using a section | 246 |
| Designing a section with a subreport | 248 |
| Designing a nested report section | 248 |
| Designing a sequential section | 249 |
| Designing a conditional section | 249 |
| Designing a parallel section | 250 |
| Sharing a connection among report sections | 251 |
| Understanding section classes | 252 |
| About AcSequentialSection | 253 |
| About a sequential section contained in a non-data section | 253 |
| About a sequential section contained in a data section | 253 |
| About customizing a sequential section | 254 |
| About AcConditionalSection | 254 |
| About AcParallelSection | 254 |
| Working with a nested report section | 256 |
| Customizing a nested report section | 256 |
| About setting an inner section's parameter in OnRow() | 256 |
| Exercise 10 | 257 |
| Using a nested report section | 258 |
| Customizing a sequential section | 262 |
| About SelectContent() | 262 |
| About NewContent() | 263 |
| Exercise 11 | 264 |
| Using a sequential section | 265 |
| Working with a conditional section | 270 |
| Customizing a conditional section | 270 |
| About ConditionIsTrue() | 271 |
| Exercise 12 | 272 |
| Using a conditional section | 273 |
| Summary | 276 |
| Chapter 11 | |
| Developing dynamic report content | 277 |
| Developing dynamic report content | 278 |
| Overview | 279 |
| About dynamic report content | 279 |
| What you learn | 279 |
| Supporting information | 279 |
| Understanding dynamic report design | 280 |
| About static report design | 280 |
| About dynamic report design | 280 |
| About the types of dynamic content | 281 |
| About dynamic queries | 281 |
| About simple report components | 281 |

| | |
|--|------|
| About structured report components. | .282 |
| Understanding dynamic queries. | .283 |
| About data row variables | .283 |
| About dynamically building a query. | .283 |
| Exercise 13 | .284 |
| Using a dynamically modified query | .285 |
| Creating a simple report component | .294 |
| About naming and scoping | .294 |
| About the framework | .294 |
| Exercise 14 | .295 |
| Dynamically creating a control. | .296 |
| Creating a structured report component | .313 |
| Designing a structured report component. | .313 |
| Using the framework. | .313 |
| Exercise 15 | .315 |
| Creating a section dynamically. | .316 |
| Summary | .326 |

Chapter 12

Using Requester API.327

| | |
|---|------|
| Using Requester API | .328 |
| Overview | .329 |
| About Requester API. | .329 |
| What you learn | .329 |
| Supporting information | .329 |
| Writing a Requester API application. | .330 |
| Writing an application that connects to iServer | .330 |
| Writing an application for local reporting | .331 |
| About the Requester API call sequence for report generation | .332 |
| About the functions for establishing and clearing the environment | .334 |
| Establishing the environment | .334 |
| About AcReqInitialize() | .334 |
| About GetAppContext() | .334 |
| About AcReqConnect() | .334 |
| Clearing the environment | .335 |
| About AcReqDisconnect() | .335 |
| About AcReqUnInitialize() | .335 |
| About the functions for generating a report | .336 |
| About AcReqSetEUDTPath() | .336 |
| About AcReqReadFile() | .336 |
| About the functions for using a parameter | .337 |
| About the functions for accessing a parameter name | .337 |
| About the AcReqGetValue functions | .337 |
| About the AcReqGetDefaultValue functions. | .337 |
| About the AcReqSetValue functions. | .338 |
| About AcReqWriteFile() | .338 |
| About AcReqGenerateReport() | .338 |
| About AcReqCloseFile() | .339 |
| About AcReqGetError() | .339 |
| About AcReqGetErrorString() | .339 |
| Using Requester API to generate a report. | .340 |
| Exercise 16 | .341 |
| Generating a report using Requester API | .342 |

| | |
|--|-----|
| About the Requester API call sequence for report viewing or printing | 347 |
| About the functions for viewing or printing an existing report | 349 |
| About AcReqSetEUPath() | 349 |
| About AcReqSetPrinterName() | 350 |
| About AcReqPrintReport() | 350 |
| About AcReqViewReport() | 350 |
| Using Requester API to display or print a report. | 351 |
| Exercise 17 | 352 |
| Displaying a report using Requester API | 353 |
| Summary | 357 |

Chapter 13

Optimizing Actuate reports359

| | |
|--|-----|
| Optimizing Actuate reports. | 360 |
| Overview | 361 |
| About optimizing a report. | 361 |
| What you learn | 361 |
| Understanding reporting requirements | 362 |
| About reporting requirements. | 362 |
| About testing report performance | 363 |
| Understanding how a query affects report performance | 364 |
| About testing query performance. | 365 |
| Improving report structure and performance. | 366 |
| Reusing components | 366 |
| Encapsulating calculations. | 366 |
| Using subreports | 366 |
| Using a multiple-input filter. | 366 |
| About group structure | 367 |
| Using a sort filter | 367 |
| Generating a structured report | 368 |
| Exercise 18 | 369 |
| Generating a report with nested report sections | 370 |
| About problems affecting a design with nested report sections | 375 |
| About optimization techniques for producing a structured report. | 376 |
| Choosing an optimization technique | 377 |
| Exercise 19 | 378 |
| Comparing structured reports | 379 |
| Examining StructuredRptMIF.rod | 380 |
| Examining StructuredRptOneQuery.rod | 385 |
| Examining StructuredRptFourGroups.rod. | 387 |
| Testing report generation time | 390 |
| About factors affecting report generation times. | 390 |
| Testing report generation time | 391 |
| Exercise 20 | 392 |
| Examining a test report | 393 |
| Running TimingTest.rox on Actuate iServer | 399 |
| Calculating report metrics | 400 |
| About other optimization techniques | 403 |
| Using data sorters. | 403 |
| Using views or stored procedures | 403 |
| Using multiple connections to data warehousing products | 403 |
| Summary | 404 |

| | |
|---------------------------|------------|
| Index | 405 |
| Quiz answers | 437 |