



Oracle JDeveloper 11g Handbook

ORACLE® 11g
FUSION MIDDLEWARE
JDEVELOPER

A Guide to Oracle Fusion Web Development

Duncan Mills

Senior Director of Product Management, Oracle

Peter Koletzke

Technical Director and Principal Instructor, Quovera

Dr. Avrom Roy-Faderman

Java EE Architect and Senior Instructor, Quovera

Foreword by Ted Farrell, Chief Architect and Senior Vice President, Tools and
Middleware, Oracle



ORACLE®

Oracle Press™

Oracle JDeveloper 11g Handbook

A Guide to Oracle Fusion Web Development

About the Authors

Duncan Mills is senior director of product management for Oracle's Application Development Tools, including Oracle JDeveloper, Oracle Enterprise Pack for Eclipse, Forms, and the ADF Framework. Duncan is currently responsible for product direction, evangelism, and courseware development around the development tools products. He has been working with Oracle in a variety of application development and DBA roles since 1988. For the past fourteen years he has been working at Oracle in both support and product development, spending the last eight years in product management. Duncan is the co-author of the Oracle Press book *Oracle JDeveloper 10g for Forms and PL/SQL Developers: A Guide to Web Development with Oracle ADF* (McGraw-Hill Professional, 2006).

For nearly three decades, Oracle, the world's largest enterprise software company, has provided the software and services that enable organizations to get the most up-to-date and accurate information from their business systems. With over 275,000 customers—including 98 of the Fortune 100—Oracle supports customers in more than 145 countries. For more information about Oracle, visit www.oracle.com.

Peter Koletzke is a technical director and principal instructor for the Enterprise e-Commerce Solutions practice at Quovera, in Mountain View, California, and has worked in the database industry since 1984. Peter has presented at various Oracle users group conferences more than 250 times and has won awards such as Pinnacle Publishing's Technical Achievement, Oracle Development Tools Users Group (ODTUG) Editor's Choice (twice), ODTUG Best Speaker, ECO/SEOUC Oracle Designer Award, the ODTUG Volunteer of the Year, and NYOUG Editor's Choice (twice). He is an Oracle ACE Director and an Oracle Certified Master. Peter has co-authored seven other Oracle Press books (variously with Dr. Paul Dorsey, Avrom, and Duncan) about Oracle JDeveloper, Oracle Forms and Reports, and Oracle Designer.

Quovera (www.quovera.com) is a business consulting and technology integration firm that specializes in delivering solutions to the high technology, telecommunications, semiconductor, manufacturing, software and services, public sector, and financial services industries. Quovera deploys solutions that deliver optimized business processes quickly and economically, driving increased productivity and improved operational efficiency. Founded in 1995, the company has a track record of delivering hundreds of strategy, design, and implementation projects to over 250 Fortune 2000 and high-growth middle market companies. Quovera's client list includes notable organizations such as Cisco Systems, ON Semiconductor, New York State, Sun Microsystems, Lawrence Livermore National Laboratory, Seagate, Toyota, Fujitsu, Visa, and Cendant.

Avrom Roy-Faderman works for Quovera as a Java EE consultant, architect, developer, and instructor, specializing in the Oracle Application Development Framework, as well as an Oracle ACE Director. He is a Sun Certified Programmer for the Java Platform and has acted as lead developer on diverse application projects ranging from e-business and online order systems to health-care provisioning for governmental and private organizations. He's also a regular speaker at Oracle user group conferences and is the co-author, with Paul Dorsey and Peter Koletzke, of two books: the *Oracle9i JDeveloper Handbook* and the *Oracle JDeveloper 10g Handbook*, both from McGraw-Hill and Oracle Press.



NOTE

Sample code for the hands-on practices in this book as well as errata are available from the websites mentioned in the Introduction's section "Websites for Sample Files."

ORACLE®

Oracle Press™

Oracle JDeveloper 11g Handbook

A Guide to Oracle Fusion Web Development

Duncan Mills
Peter Koletzke
Avrom Roy-Faderman

**Mc
Graw
Hill**

New York Chicago San Francisco
Lisbon London Madrid Mexico City Milan
New Delhi San Juan Seoul Singapore Sydney Toronto

Copyright © 2010 by The McGraw-Hill Companies, Inc. All rights reserved. Except as permitted under the United States Copyright Act of 1976, no part of this publication may be reproduced or distributed in any form or by any means, or stored in a database or retrieval system, without the prior written permission of the publisher.

ISBN: 978-0-07-160239-6

MHID: 0-07-160239-9

The material in this eBook also appears in the print version of this title: ISBN: 978-0-07-160238-9, MHID: 0-07-160238-0.

All trademarks are trademarks of their respective owners. Rather than put a trademark symbol after every occurrence of a trademarked name, we use names in an editorial fashion only, and to the benefit of the trademark owner, with no intention of infringement of the trademark. Where such designations appear in this book, they have been printed with initial caps.

McGraw-Hill eBooks are available at special quantity discounts to use as premiums and sales promotions, or for use in corporate training programs. To contact a representative please e-mail us at bulksales@mcgraw-hill.com.

Information has been obtained by Publisher from sources believed to be reliable. However, because of the possibility of human or mechanical error by our sources, Publisher, or others, Publisher does not guarantee the accuracy, adequacy, or completeness of any information included in this work and is not responsible for any errors or omissions or the results obtained from the use of such information.

Oracle Corporation does not make any representations or warranties as to the accuracy, adequacy, or completeness of any information contained in this Work, and is not responsible for any errors or omissions.

TERMS OF USE

This is a copyrighted work and The McGraw-Hill Companies, Inc. (“McGraw-Hill”) and its licensors reserve all rights in and to the work. Use of this work is subject to these terms. Except as permitted under the Copyright Act of 1976 and the right to store and retrieve one copy of the work, you may not decompile, disassemble, reverse engineer, reproduce, modify, create derivative works based upon, transmit, distribute, disseminate, sell, publish or sublicense the work or any part of it without McGraw-Hill’s prior consent. You may use the work for your own noncommercial and personal use; any other use of the work is strictly prohibited. Your right to use the work may be terminated if you fail to comply with these terms.

THE WORK IS PROVIDED “AS IS.” MCGRAW-HILL AND ITS LICENSORS MAKE NO GUARANTEES OR WARRANTIES AS TO THE ACCURACY, ADEQUACY OR COMPLETENESS OF OR RESULTS TO BE OBTAINED FROM USING THE WORK, INCLUDING ANY INFORMATION THAT CAN BE ACCESSED THROUGH THE WORK VIA HYPERLINK OR OTHERWISE, AND EXPRESSLY DISCLAIM ANY WARRANTY, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. McGraw-Hill and its licensors do not warrant or guarantee that the functions contained in the work will meet your requirements or that its operation will be uninterrupted or error free. Neither McGraw-Hill nor its licensors shall be liable to you or anyone else for any inaccuracy, error or omission, regardless of cause, in the work or for any damages resulting therefrom. McGraw-Hill has no responsibility for the content of any information accessed through the work. Under no circumstances shall McGraw-Hill and/or its licensors be liable for any indirect, incidental, special, punitive, consequential or similar damages that result from the use of or inability to use the work, even if any of them has been advised of the possibility of such damages. This limitation of liability shall apply to any claim or cause whatsoever whether such claim or cause arises in contract, tort or otherwise.

For Amanda, Josh, Sam, and Hissy.
—Duncan Mills

For my folks, Jan and Max.
—Peter Koletzke

For Nilakash, a sudden blue sky on a rainy day.
—Avrom Roy-Faderman

This page intentionally left blank



Contents at a Glance

PART I Overviews

1	Overview of Fusion Development and ADF	3
2	Introduction to the JDeveloper Integrated Development Environment	33
3	JDeveloper Tools	61
4	Technology Background	105

PART II ADF Business Components

5	Introducing ADF Business Components	147
6	Specifying a Business Domain	169
7	Assembling a Data Model	233
8	Providing Custom Data Services	285

PART III ADF View and Controller

9	JavaServer Faces Basics	309
10	Your First JSF	327
11	The ADF Controller and Task Flows	371
12	ADF Faces Rich Client	413

PART IV
ADF Model

13	ADF Model Basics	455
14	ADF Advanced Bindings	489
15	ADF Bindings: Iterators, Executables, and Code	537

PART V
Developing the Sample Application

16	Application Design Principles and Sample Application Overview	567
17	Sample Application: UI Patterns and Templates	597
18	Sample Application: Search Page and Department Hierarchy Page	627
19	Sample Application: Edit Page	667
20	Sample Application: The Image LOV Dialog	723
21	Sample Application: Security	773
22	Sample Application: Deployment	799
	Index	821



Contents

Foreword	xxi
Acknowledgments	xxiii
Introduction	xxix

PART I **Overviews**

1	Overview of Fusion Development and ADF	3
	What Is Fusion?	4
	The Origins of Oracle Fusion Architecture	5
	The Fusion Architecture	13
	What Is a Framework?	17
	The Anatomy of a Framework	18
	What Characterizes a Good Framework?	20
	What Is Oracle ADF?	22
	How Do I Choose Technologies?	23
	User Interface Technology	24
	Database Integration	27
	Why Should I Use JDeveloper?	29
2	Introduction to the JDeveloper Integrated Development Environment	33
	What Are the Important Features of the JDeveloper IDE?	35
	JDeveloper Roles	36
	The Main IDE Window	36
	Main Toolbar	46
	Main Menu Bar	46
	How Can I Best Use the Help Center?	55
	Finding the Help Topic	55
	How Does Source Code Version Control Work in JDeveloper?	56
	Working with Subversion	57
	Committing Files	58

X Oracle JDeveloper 11g Handbook: A Guide to Oracle Fusion Web Development

Adding Files	59
Refreshing a Working Copy	59
Version History and Version Compare	60
Version Navigator	60
State Overlay Icons	60
Team Productivity Center	60
3 JDeveloper Tools	61
Application Navigator	62
Projects Panel	63
Application Resources Panel	66
Data Controls Panel	67
Recently Opened Files Panel	67
Resource Palette	67
New Gallery	68
Wizards	70
Creation Dialogs	71
Application Overview Window	72
Application Overview Panel Techniques	73
How Can the Application Overview Window Help?	74
Structure Window	74
Structure Window Operations	74
Structure Window Views	76
Component Palette	77
My Components	77
Property Inspector	78
Property Inspector Operations	79
Log Window	80
Code Editor	81
End Tag Completion and Code Completion	82
Code Templates	82
Code Insight and Tag Insight	83
Other Text Editing Features	84
Debugger	90
Debugger Windows	92
Running the Debugger	93
Debugging PL/SQL	93
Other Tools for Improving Code	93
JSP/HTML Visual Editor	94
Modelers and Diagrammers	95
Types of Diagrams	95
Creating a UML Diagram	96
Activity Diagram	98
XML Schema Editor	99
JSF Navigation Diagrammer	100

	ADF Task Flow Diagrammer	100
	Struts Page Flow Modeler	100
	Database Navigator	100
	SQL Worksheet	101
	Database Documentation	103
4	Technology Background	105
	What Is Java Platform, Enterprise Edition?	106
	Online References	107
	Java EE Architecture Model	108
	BluePrints and Design Patterns	110
	What Are the Main Styles of Java EE Code?	111
	Application Client	112
	Web Client	114
	How Do Web Communications Work?	118
	HTTP Overview	118
	HTTP Request	119
	HTTP Response	120
	Methods	122
	The Steps in a Web Application Round-Trip	123
	Asynchronous JavaScript and XML (AJAX)	125
	What Languages Do I Need to Know for Web Development?	128
	The Web Development Experience	128
	The Level of Knowledge Needed	129
	Java	129
	Hypertext Markup Language (HTML)	132
	JavaScript	135
	Cascading Style Sheets (CSS)	137
	Extensible Markup Language (XML)	139
	Expression Language	140
	Groovy	141

PART II

ADF Business Components

5	Introducing ADF Business Components	147
	Why Use ADF BC?	148
	Java Enterprise Design Patterns	148
	Business Logic Framework	149
	Deployment Flexibility	150
	Database Orientation	150
	How Do Business Components Use Java and XML?	151
	What Are the Major Business Components?	152
	Entity Object Definitions	153
	Associations	156
	View Object Definitions	157

View Link Definitions	160
Application Module Definitions	162
Which Components Support the Major Business Components?	164
Domains	164
Validation Rules	165
Business Logic Units	165
Property Sets	165
How Can I Visualize My Business Components?	166
6 Specifying a Business Domain	169
What Are Entity Object Definitions and Entity Object Instances?	170
Entity Object XML Metadata	170
Entity Object Classes	171
Entity Definition Classes	173
Entity Collection Classes	174
Custom Framework Classes	174
What Is an Association?	176
Cardinality	177
Source and Destination Attributes	179
Association Accessors	180
Compositions	181
How Do Entity Attributes Work?	182
Persistent and Transient Attributes	182
Attribute Types and Column Datatypes	183
Working with Built-in Domains	184
Object Type Domains	185
Primary Key and Alternate Key Attributes	185
How Do You Create and Edit Entity Object Definitions and Associations?	186
Creating Entity Object Definitions and Associations	186
Editing Entity Object Definitions and Associations	188
How Do You Implement Attribute-Level Validation?	190
Specifying Mandatory Attributes	191
Specifying Scale and Precision	192
Preventing Users from Changing Attribute Values	192
Attribute-Level Validation Rules	192
Adding Validation to Setter Methods	202
Validation Domains	204
How Do You Implement Entity-Level Validation?	206
Entity-Level Validation Rules	207
Validation in the Entity Object Class	211
How Does the ADF BC Validation Cycle Work?	212
The Attribute-Level Validation Process	212
The Nondeferred Entity-Level Validation Process	213
The Deferred Entity-Level Validation Process	214
Validation Loops	214

How Do You Implement Attribute Defaulting and Calculation?	215
Attribute Defaulting	216
Attribute Calculation	217
How Do You Synchronize with Database Defaulting and Calculation?	220
The DBSequence Domain	221
Other Database Defaulting and Database Calculations	228
How Do You Implement Polymorphic Business Rules?	228
Business Logic Groups	228
Polymorphic Entity Object Definitions	231
7 Assembling a Data Model	233
What Are View Object Definitions, View Object Instances, and View Rows?	234
View Object XML Metadata	235
View Object Classes	235
View Row Classes	236
View Definition Classes	237
Custom Framework Classes	237
How Do View Attributes Work?	238
Persistent, SQL-Only, Entity-Derived, and Transient View Attributes	238
SQL-Only View Object Definitions	244
View Attribute Updatability and Read-Only Entity Object Usages	244
View Attribute Types	245
Key and Alternate Key Attributes	245
How Do You Create and Edit View Object Definitions?	246
Creating View Object Definitions	246
Editing View Object Definitions	247
How Do You Specify View Object Queries?	249
Normal SQL Mode	250
Declarative SQL Mode	254
Expert SQL Mode	257
Dynamic Queries	259
How Can View Object Instances Handle Row Creation, Row Deletion, and Association Changes?	261
Reference Entity Usages	261
Creating View Rows	262
Deleting View Rows	263
What Are View Link Definitions and View Link Instances?	263
Source and Destination Attributes	263
View Link SQL	264
View Link Accessors	266
How Do You Create and Edit View Link Definitions?	267
Creating View Link Definitions	267
Editing View Link Definitions	268
What Are Application Module Definitions and Application Module Instances?	268
Application Module XML Metadata	268
Application Module Classes	269

Application Module Definition Classes	270
Custom Framework Classes	270
How Application Module Instances Implement Their Data Model	271
How Do You Create and Edit Application Module Definitions?	271
Creating Application Module Definitions	271
Editing Application Module Definitions	273
Editing or Creating Configurations for Application Module Definitions	275
How Do You Use View Object Instances as LOVs?	277
View Accessors	277
LOV View Attributes	279
Static View Object Definitions	281
How Do Application Module Instances Manage Transactional State?	282
Locking	282
Application Module Pooling	283
8 Providing Custom Data Services	285
What Sort of Method Can Be a Service Method?	287
How Do You Expose Methods as Service Methods?	288
Client Interfaces	288
Adding Methods to Client Interfaces	289
How Do You Implement Single-Row Methods?	290
How Do You Implement Query-Level Methods?	292
Retrieving View Rows	292
Navigating Through the Cache	295
Creating and Deleting Rows	296
Modifying Queries and Filtering Rows	297
How Do You Implement Cross-Query or Transaction-Level Methods?	299
Finding View Object and Nested Application Module Instances in the Data Model	299
Creating and Removing View Object Instances Dynamically	300
Creating and Removing Master-Detail Relationships Dynamically	302
Manipulating Transactions	305

PART III

ADF View and Controller

9 JavaServer Faces Basics	309
Why Was JSF Developed?	311
What JSF Concepts Do I Really Need to Know Now?	312
What Is JSF?	313
Runtime Architecture	313
JSF-Oriented Files	317
Components	320
Designing Cross-Platform Interfaces	324
Where Can I Find Additional Information about JSF?	325
Websites	326
Printed Books	326

10	Your First JSF	327
	Hands-on Practice: Create a JSF Login and JSF Home Page	328
	I. Create the Application and a Project	330
	II. Diagram the Page Flow	334
	III. Create the Login and Home Pages	336
	IV. Add Components to the Login Page	344
	V. Add Components to the Home Page	354
	VI. Add Navigation and Validation Logic to the Application	359
	Centralize the Messages	365
	Hands-on Practice: Access a Message Bundle with EL	366
	Hands-on Practice: Access a Message Bundle with Java	368
11	The ADF Controller and Task Flows	371
	What Is the ADF Controller and Why Is It Needed?	372
	The Problem with JSF	372
	Creating a Task Flow Definition	373
	About Task Flows and Managed Beans	374
	Task Flow Components	377
	Types of Task Flows	389
	Unbounded Task Flows	389
	Bounded Task Flows	390
	Pages or Fragments?	395
	More Options on Task Flow Creation	397
	Advanced Task Flows	402
	Data Control Scope	402
	Transaction Management	403
	The Back Button	404
	Coding Around a Flow	404
	Debugging Task Flows	407
12	ADF Faces Rich Client	413
	What Is ADF Faces and Where Did It Come From?	414
	HTML Render Kit	415
	A Very Brief History	416
	ADF Faces Is Better	416
	How Does Partial Page Rendering Work?	417
	Hands-on Practice: Demonstrate Basic PPR	418
	How Can I Achieve the Perfect Layout?	421
	Facets	421
	Knowledge of Layout Containers	422
	Nesting Layout Containers	426
	Hands-on Practice: Implement an Across and Down Tab Order	427
	Quick Start Layouts	432
	What Types of Atomic Components Are Available?	433
	Other Atomic Components	436
	Data Visualization Tools (DVT)	440

How Can I Change the Default ADF Faces Appearance?	446
Working with Skins	446
Working with Component-Level Styles	449
How Will I Ever Learn about All These Components?	451

PART IV
ADF Model

13	ADF Model Basics	455
	What Is the ADF Model Layer?	456
	Data Controls	457
	Binding Containers	457
	How Do You Provide Default or Dynamic Visual Properties for UI Controls?	458
	Control Hints for Attributes	458
	Control Hints for Bind Variables	462
	Control Hints for Lists of Values	462
	How Do You Create Data-Bound Pages?	464
	Using the Data Controls Panel with Pages	466
	“Drop As” Options	467
	How Do You Create Data-Bound Activities?	485
	What Files Are Involved in Data Binding?	486
14	ADF Advanced Bindings	489
	How Can I Edit Data Bindings?	491
	Cleaning Up Bindings	492
	Exploring the Binding Categories	492
	What Binding Types Are Available?	494
	Attribute Bindings	495
	List Bindings	498
	List of Values Bindings	508
	Tree Bindings	514
	Table Bindings	522
	Action Bindings	524
	Method Bindings	532
	Boolean Bindings	534
15	ADF Bindings: Iterators, Executables, and Code	537
	What Is an Iterator?	538
	Editing the Iterator	538
	Controlling Iterator Execution	540
	Types of Iterators	540
	What Are Executables?	541
	invokeAction Executable	541
	taskFlow Executable	542
	page Executable	544
	searchRegion Executable	544
	shuttleRegion Executable	545

How Can I Control Execution in the ADF Model?	546
Another Look at Executables	546
How Can I Override and Add to the ADF Model Life Cycle?	548
The ADFm Life Cycle	549
Customizing Error Handling	553
What Are Parameters?	554
Populating PageDef Parameters	554
Advanced PageDef Parameter Properties	555
What Are Contextual Events?	555
A Worked Example of Contextual Events	556
How Can I Write Code Using the ADF Model?	559
Access to Data and Methods	559
Executing Methods and Operations	562

PART V

Developing the Sample Application

16 Application Design Principles and Sample Application Overview	567
What Is the Design Process for a Fusion Application?	569
System Design Tools and the SDLC	569
General Application Design Considerations	572
Typical Fusion Page Design	575
Approaching ADF Business Components Design	576
What Best Practice Should Be Adopted?	579
Use Framework Buffer Classes	579
Use Java Packaging	581
Stick to JSF and Only JSF	581
Create a Custom Skin	581
What Is the Design of the Sample Application?	582
Database Design	582
Application Design	584
What Do I Have to Think about Before Starting?	587
Application Templates	587
Source Control?	590
Unit Testing	593
Auditing and Code Analysis	594
17 Sample Application: UI Patterns and Templates	597
I. Prepare the Application Workspace	599
II. Prepare the Database	602
Create a Database Diagram	603
Modify the Diagram	606
Alter the Database	610
III. Set Up Project Defaults	612
Create Model Framework Superclasses	612
Set Up ViewController Properties	613
Declare the Application's Skin	614

IV. Define a JSF Page Template	615
Create the File	615
Add Boxes and Panels	617
Define a Menu and Popup	621
Define the Named Content Area of the Template	623
Test the Template	623
Fix the Copyright Text	625
18 Sample Application: Search Page	627
I. Prepare Employee Search Business Components	629
Set Up Entity Object Definitions	630
Create the Employees View Object Definition	632
Define Named View Criteria	633
II. Create the Search Page	635
III. Create the Employee Search Area	637
Define the Search Task Flow and Page Fragment	637
Refine the Layout	640
Add an Export Capability to the Result Set	644
Add an Employee Details Popup	645
Add an In-Process Indicator	646
IV. Create the Hierarchy Viewer Area	648
Set Up Hierarchy Node View Object Definitions	648
Create the Hierarchy Task Flow and Page Fragment	651
Explore the Page	652
Refine the Hierarchy Viewer	654
Improve the Department Node Display	657
Improve the Employee Node Display	659
Plug the Hierarchy Viewer into the Page	662
Write a UI Manager to Switch Page Fragments	662
19 Sample Application: Edit Page	667
I. Set Up the Edit Page and Add LOVs	669
Create the Edit Page Fragment	669
Add the Department List of Values	673
Add the Job List of Values	674
Add the Manager List of Values	675
II. Refine the Salary Field	682
Add Validation to the Salary Field	682
Fix the Salary Validation and Change the Salary Format	686
III. Refine the Commission Field	688
Add Validation to the Commission Field	688
Change Commission to a Spin Box	690
Change the Commission Updatable Property	691
Clear the Commission Value for Non-Sales Employees	694

IV. Refine the Image Field and the Layout	697
Display the Employee Image	697
Rearrange the Layout and Modify Some Labels	698
Fix the Forms and the Field Widths	700
V. Create a Popup for Job History	703
Display the Job History in a Popup	704
Update the Job History	707
VI. Apply Final Refinements	713
Define Cascade Delete Functionality	713
Add the Biography	714
Disable and Enable the Rich Text Editor	717
Refactor the getBindings() Method	719
Display the Manager Name	720
20 Sample Application: The Image LOV Dialog	723
I. Change How the Business Components Handle Image Usages	727
Create an Entity Hierarchy for Image Usages	727
Use the Entity Hierarchy in a View Object Definition	729
II. Create the Popup Dialog	733
Create a Task Flow for the Dialog	733
Create a Page Template for the Dialog	734
Create Page Fragments for the Train Stops	736
Wire the Dialog to the Edit Page	736
Fix the Popup Behavior	738
III. Implement the Add Image Page	744
Lay Out the Form Components	744
Create a Backing Bean for the Add Image Page	745
Use the Image Component to Preview the Uploaded Image	745
Create a Service Method to Add an Image	747
Fix Association Posting Behavior	751
Create an Upload Listener to Copy the File and Call the Service Method	752
IV. Implement the Set Default Image Page	756
Lay Out the Form Components	756
Create a Backing Bean for the Set Default Image Page	759
Add Drag-and-Drop Capabilities	759
Create a Service Method to Set the Default Image	760
Implement a Drop Listener to Call the Service Method	762
Add Commit Functionality	765
V. Integrate the Edit Page with the Rest of the Application	766
Create a Link from the Search Page to the Edit Page	766
Parameterize the Edit Employee Task Flow	768
Restrict the EditEmployee View Object Instance's Query	
Using the Parameters	768
Pass Parameters to the Edit Employee Task Flow	770