ADF Code Corner

73. Hands on – Creating a search form using a POJO WS and the Web Service Data Control



Introduction

The Web Service Data Control is one of two options ADF developers have to access JAX-WS Web Services. It exposes service collections, methods and attributes in the Oracle JDeveloper Data Controls panel for declarative use. In this end-to-end hands-on exercise you will

- create a Web Service from a JavaBean class
- configure the ADF WS Data Control
- define UI hints and validation.

A screenshot of the application runtime is shown below.

In the sample application, you use an ADF form to query data from a Web Service, which exposes a search function accepting a complex argument of type Employee. The result set is displayed in the table without the need to refresh the whole page.

Attp://127.0	.0.1:7101/WsSamp	le-ViewControl	ler				<u>`</u> •	2 - 🖂
Department Id		50	Hire Da	ate	20	Phone		
Mail			Job	Id	ST_	Salary		
Employee Id			Last Nar	me	Ki	Phone		
First Name			Manager	Id		Salary		
Submit								
Submit	Mail	Employee I	id I	First Name	Hire Date	Job Id	Last Name	Manage
Submit Department Id 50	Mail IMIKKILI	Employee I 126	id I	First Name	Hire Date 9/28/1998	Job Id ST_CLERK	Last Name Mikkilineni	Manage 120

Prerequisite and Setup

The POJO model uses hard coded data values that match the data of the Oracle HR demo schema. There is no database required to run this hands-on example. You need Oracle JDeveloper 11g 11.1.1.3 or later installed to follow this hands-on.

Before you start, download and extract the "WsSampleBase.zip" starter workspace. Open Oracle JDeveloper and navigate to the directory where you extracted the ZIP file and open the WsSampleBase.jws workspace file.

The initial Oracle JDeveloper IDE view is shown below



Note: In this hands-on, the Web Service project and the ADF model and view controller project are within a single JDeveloper workspace. In reality, the Web Service would be deployed on a remote server. We simplified the setup for this hands-on so you can use the integrated WebLogic Server for the Web Service runtime too.

Creating the POJO Web Service



1. To create the Web Service, expand the POJO Model project's oracle.pojo.facade package and select the SessionFacade.java class.

2. From the right mouse context menu, choose Create Web Service

3. In the Web Service creation dialog, choose the "Java EE 1.5, with support for JAX-WS Annotations" option to build an annotation based Web Service from the POJO model.



4. In the "Web Service Name" field, type "HrPojoWsService". The WS name automatically becomes the name for the WS port too. The WS port is later used when accessing the Web Services from ADF.

🖕 Create Java Web Service	- Step 3 of 11
Generation Options	0101010101010101010101010
A Welcome	Select the Java dass or Stateless Session EJB you want to publish, and the J2EE and annotations options for web service.
Select Deployment Platfi	Component To Publish:
Generation Options	oracle.pojo.facade.SessionFacade
 Service Endpoint Interfa 	
Message Format	Web Service Name:
A Methods	HrPojowsService
O Choose Service State	Port Name
🔍 Additional Classes	HrPojowsPort
Configure Policies	Add SEI
Provide Handler Details	
0 Finish	
Help	< <u>Back</u> <u>Next</u> <u>Finish</u> Cancel

5. Click Next

6. Again, click "Next" and accept all the default settings in Step 4 of the WS Creation wizard

💩 Create Java Web Service	- Step 4 of 10
Message Format	0101010101010101010101
A Welcome	Use the controls on this page to choose the binding option and the settings that control the structure of the SOAP messages transmitted to and from the web service.
Select Deployment Platfi	SOAP 1.1 Binding
Generation Options	SOAP 1.2 Binding
😠 Message Format	
Methods	SOAP Message Format: Document/Wrapped
Choose Service State	
Additional Classes	Binary encoding style for this service is MTOM. Enable MTOM to write MTOM apportation for
Configure Policies	this service.
Provide Handler Details	Enable MTOM
O Finish	
Help	< <u>Back</u> <u>Next</u> > <u>Einish</u> Cancel

7. The POJO class exposes more methods than needed for this hands-on. Press "Deselect All" and then select all methods that have a "findEmployee" in the name.



8. Click "Finish" to generate the Web Service definition

9. In the Oracle JDeveloper Application Navigator, select the "SessionFacade.java" entry, which has its icon changed to indicate the Web Service functionality, and choose "Test Web Service" from the right mouse context menu.



Note: If this is your first time experience with building JAX-WS Web Services, you may double click onto the Java class entry to see the added annotations.

10. The WS tester shows the Web Service WSDL file reference, a list box exposing Web Service functions, a form to input request parameters and an area to display the Web Service response

HTTP An	alyzer : Unsent Message					
8 🗿 🔺	- 🖂 📋 🖪 🏟					
URL:	http://localhost:7101/WsSamp	vle-PojoModel-context-root/HrPojoWsPort				-
WSDL URL:	http://localhost:7101/WsSampl	le-PojoModel-context-root/HrPojoWsPort?WSDL				Open WSDL
Operations:	HrPojoWsPort.findEmployeesB	3yLastName(,)		Credentials:	<no credential=""></no>	• <u>N</u> ew
A.						
	HTTP Headers	G	+ · × ¦	Response HT	TTP Headers	
🗄 SOAP He	aders			Request is being (edited	
🖃 paramet	ers					
arg0 : :	string	✓ Include				
Send	Request Copy Reque	lest				

11. Click the "Open WSDL" button to open the "Select Web Service Decsription" dialog.

12. Mark the WSDL string reference and copy it to the clipboard using the ctrl+c keyboard key. The WSDL reference is needed when creating the WS Data Control. For this hands-on, the Web Service is deployed to the integrated WebLogic Server in Oracle JDeveloper.

Select Web S	ervice Description		×
Enter the loc	ation of the WSDL document describing the service that you wish to	invoke.	
WSDL URL:	http://localhost:7101/WsSample-PojoModel-context-root/HrPojoW	/sPort?WSDL	Browse
Help		ОК	Cancel

13. Ok the dialog

14. From the "Operations" list, select "HrPojoWsPort.findEmploye(,)", which is a function that expects an Employee object type as an input argument

15. In the parameters form, select the "departmentId" attribute to be included in the request and unselect all the other options. Define a value of "60" for the departmentId attribute (no quotes)

16. Press the "Send Request" button

	http://localhost:71	01/WsSample-PojoModel	-context-root/HrPojoWsPo	rt			
URL:	http://localhost:71	01/WsSample-PojoModel	-context-root/HrPojoWsPo	rt?WSDL			
ations:	HrPojoWsPort.find	HrPojoWsPort.findEmployee(,)					
equest I	HTTP Headers			÷.			
AP Hea	aders						
arg0				💌 Re			
depa	rtmentId : long	60		 Include 			
email	: string			Include			
emplo	oyeeId : long			Include			
firstN	lame : string			Include			
hireD	ate : dateTime			Include			
jobId	: string			Include			
lastna	ame : string			Include			
mana	gerId : long			Include			
phon	eNumber : string			Include			
salar	y : long			Include			

17. The WS response is displayed in the response area

sponse HTTP Headers	٩	🖉 200 C
arameters		
return : Array		
🖃 return		
departmentId	60	
email	AHUNOLD	
employeeId	103	
firstName	Alexander	
hireDate	1990-01-03T00:00:00+01:00	
jobId	IT_PROG	
lastname	Hunold	
managerId	102	
phoneNumber	590.423.4567	
salary	9000	
🖃 return		
departmentId	60	
email	BERNST	
employeeId	104	
firstName	Bruce	
hireDate	1991-05-21T00:00:00+02:00	
jobId	IT_PROG	
lastname	Ernst	
managerId	103	
phoneNumber	590.423.4568	

18. Close the WS tester.

Creating the WebService Data Control

To create the Web Service Data Control definition, a WSDL reference is needed for JDeveloper to write the service structure into XML metadata.

1. Select the "WsDcModel" project and choose "New" from the right mouse context menu to create a new WS Data control configuration



2. In the New Gallery, select the Business Tier | Web Service entry

3. Select the "Web Service Data Control" entry in the "Items" area and "Ok" the dialog

New Gallery Al Technologies Current Project Te This list is filtered according to the curr Search Current Project Technol	chnologies ent project's <u>selected technologies</u> . ogles
Categories:	Items: Show All Descriptions
General Connections Connections Oeployment Profiles Diagrams External Applications Java Projects ByM Tier Business Tier Data Controls Web Services SOA Tier All Ttems	Java Web Service Java Web Service from WSDL Java Web Service from WSDL PL/SQL Web Service Web Service Proxy WSDL Document JAX-RPC Mapping File UDDI Registry Connection URL Service Data Control Web Service Data Control Creates a new data control from a Web service definition.
Help	OK Cancel

ADF CODE CORNERHands on – Creating a search form using a POJO WS and
the Web Service Data Control

- 4. Type "WsSampleBaseDC" in the "Name" field to define the display name of the Data Control
- 5. In the URL field, paste the previously WSDL reference you copied when testing the Web Service.

6. Press "Next" to continue the configuration and to validate the WSDL access

💩 Create Web Service Data	Control - Step 1 of 5
Data Source	
Data Source Data Control Operations	Specify a name and service description details for the data control.
Response Format	Name: WsSampleBaseDC
C Finish	Specify a WSDL URL or a Java source file with JAX-WS annotations that describes the service. Click the Services button to display the services described.
	URL: [ttp://localhost:7101/WsSample-PojoModel-context-root/HrPojoWsPort?WSDL] Browse UDDI Services
	Select the service to create the data control.
<>	
Help	< <u>Back</u> <u>Next</u> Einish Cancel

7. From the 4 employees search functions exposed by the WS, only the "findEmployee" should be exposed by the Data Control.

🖕 Create Web Service Data Control - Step 2 of 5					
Data Control Operatio	ons	01010101010101010101010101010101010101			
 <u>Data Source</u> Data Control Operati 	Select the operations that you want the Available:	e data control to support. Selected:			
Response Format Endpoint Authentication Finish	VsService ojoWsPort findAllEmployeesByDepartmentId findEmployeesByLastName	HrPojoWsService			
	FindEmployee	8 8 8			
		< Back Next Einish	Cancel		

8. Press Next

9. In the Response Format dialog, accept the default format, which is XML. Press "Next"

👌 Create Web Service Data	Control - Step 3 of 5	X
Response Format		
Data Source Data Control Operations Response Format Endpoint Authentication Finish	Specify the format of the SOAP respon Available Operations: HrPojoWsService HrPojoWsPort ImdEmployee	se. Format Detail Fogmat: XML ▼ XSD URL: XSL URL:
Help		< Back Next Finish Cancel

10. The last step in the configuration (not shown as a screen shot) allows you to define authentication for the Web Service access. Ignore the dialog and press "Finish" to generate the WS DC configuration metadata

Application	Run Mana		R,		
🕒 WsSampleBase				•	- 🖻 -
✓ Projects			<u>a</u> 🗞	7-35	-
PojoModel ViewControl ViewControl O ViewControl O O ViewControl O O O O O O O O O O	er on Sources sample DataControls.dcx WsSampleBaseDC. findEmployee Image Return.xml findEmployee.xml ampleBaseDC.find ampleBaseDC.find findEmployee.xml A-INF adfm.xml	xml BaseDC rations.x Employee es	ml 2.params		ß
 Application Resources Data Controls 	ces			65	2
Recently Opened	Files			00	u I
, internet					

11. Expand the WSDcModel project structure and select the "Return.xml" configuration under the "findEmployee" function. The "Return.xml" document describes the result set of the WS response.

12. Open the Structure Window (ctrl+shift+S) and the Property Inspector (ctrl+shift+I)

In the Structure Window, select one of the exposed attributes to edit its "Label" and "Tool Tip" attribute. The WS Data Control allows developers to define UI Hints that are looked up by the view layer at runtime. This not only allows you to define consistent label and control hints, it also allows you to define then so they can be translated.

13. Provide "Label" and "Tool Tip" entries for the attributes as shown in the image below. You can detach the Property Inspector window and move it closer to the Structure Window using the mouse and folding the ctrl-key pressed.

Note: The property settings are saved when you step out of the property field. So make sure you briefly select another property before moving on to edit another attribute.



14. Select the WSDcModelBundle.properties file and double click it to open. The file defines keys and strings you provided for the attributes.

Though not needed for this hands-on, to create a translated version, you duplicate the file and define the name of the duplicate as WSDcModelBundle_<lang>.properties, where <lang> should be replaced with "de" for German, "fr" for French etc.

Application	▶ Run Mana		3 6	_	WsD	cModelBundle.properties
🔁 WsSampleBase			-	- 1	💏 🕶 F	ind 🗸 🖓
✓ Projects		9	® 7 - ≌	→		#
Projekts Projekts	iler tol Sources .sample DataControls.dcx WsCanpleBaseDC.v f.sample.WsSampleBaseDC.v findEmployee @Return.xml findEmployee.Oper findEmployee.xml SampleBaseDC.findE arg0.xml	roperties ml asseDC ations.xml mployee.pa	(M) Y → 2≥	-		<pre># # # # # # # # # # # # # # # # # # #</pre>
• • • • • • • • • • • • • • • • • • •	SampleBaseDC.Type findEmployee.xml	s				adf.sample.WsSampleBaseDC.findEmployee.Return.hireDate_TOOLTIP=Start Date

15. To do the same for the input arguments expected by the findEmployee, function, select the "arg0.xml"entry in the "WsSampleBase.findEmployee.params" package

16. Double click the arg0.xml entry to open the edit dialog. Note that this is a second option to define attribute of the WS DC



17. Select an attribute, e.g. "departmentId", and press the "pencil icon" to bring up the edit dialog

Note: the labels you define for the "attributes will later show in the search form. The previously edited labels are those that will show in the result table

arg0.xml ttributes ccessors perations	Attributes Name Genal Genal	Edit Attribute Search Control inits Validation Custom Properties	Type	Default Value
		Help		Apply OK Cancel

18. Edit the "Label Text" and "Tool Tip" text.

Note that there is a "Validation" section in this dialog that allows you to define validation rules for attributes that are then enforced by the ADF binding layer. We have an example of how to use it at the end of this hands-on.

19. Save the JDeveloper workspace when you are done.

Creating the Search Form

1. Select the "ViewController" project and choose "New" from the right mouse context menu.



ADF CODE CORNER Hands on – Creating a search form using a POJO WS and the Web Service Data Control

- 2. In the New Gallery, select the Web Tier | JSF category and choose the "JSF Page" option.
- 3. Press "Ok"

🕹 New Gallery		×							
All Technologies Current Project Technologies This list is filtered according to the current project's <u>selected technologies</u> . Image: Search Current Project Technologies									
Categories:		Show All Descriptions							
	 ADF Task Flow ADF Task Flow Based on Human Task ADF Task Flow Template 3SF Dedarative Component JSF Page Launches the Create JSF Page dialog, in which you create JSF Page dialog, in which you create JSF Page flow and Configuration (faces-config.xml) JSF Page Flow and Configuration (faces-config.xml) JSF Page Fragment SF Page Template 	te a new skeleton within a project in the							
<u>H</u> elp	·	Cancel							

4. In the Create JSF Page dialog define the "File Name" as "EmployeeSearch.jspx"

5. Select the "Quick Start Layout" Radio Button and press the "Browse" button

🛬 Create JSF Page		22	
Enter the name, directory, and choose a type for the J a <u>Quick Start Layout</u> to add and configure an initial set Elle Name: EmployeeSearch.jspx Directory: Ion Factory/ADF Training Chapters\15 Serv IV Create as XML Document (*.jspx) Render in Mobile Device Initial Page Layout and Content Dilang Page	Component C Categories One Column Two Column	Ivpes	×
Page Template Oracle Three Column Layout Ouck Start Layout One Column (Stretched) Browse Page Implementation (UI components are not expose Help Messages - Log	Three Column		Options □ Apply Themes One Column Header (Stretched with Splitter) Child component will this container. Dimension does not way based on browser dimension. Splitter with collapse direction. Scrollable panel.
	Help		Cancel

6. Select a one column layout as shown in the image above. This will create a page with a panel splitter component to separate out the two view areas

7. Press "Ok" on both dialogs to render the empty page

8. Expand the "Data Controls" panel

9. Select the "arg0" entry under the findEmployee_parameters node. This node resolves the attribute structure of the complex input type – and instance of the Employee object.

10. Drag the "arg0" node to the top area of the JSF page



11. Drop the "arg0" node as "Form | ADF Form" to generate the search form

EmployeeSearch.jspx								
🔞 🔹 Show 🛪 Full Screen Size 💌 🔕 None	▼ Default	- I	lone 1	- 🛝	ا 🗞	P 🖪	1	U i
	••••••							••••
								- 3
1								- 3
	Create		1					- 3
	Eorm	•	88 A	DF F <u>o</u> r	m			ĺ
1	Master- <u>D</u> etail	•	8 8	DF Reg	d-on	Forn	n	
	Single Selection	•						- 8
	Cancel							
								* ***{

12. In the opened dialog, check the "Include Submit Button" option and "Ok" the dialog

Display Label	Value Binding departmentId e email e employeeId firstName inireDate jobId lastname managerId phoneNumber salary	Component To Use ADF Input Text w/Label ADF Input Text w/Label ADF Input Text w/Label ADF Input Text w/Label ADF Input Date w/Label ADF Input Text w/Label	 ○ ○ ○ ○ ○ ○ ○ ○

13. To create a multi column form layout, select the af:panelFormLayout component in the Structure Window. The Panel Form Layout component is contained in the first facet of the af:panelSplitter component.

14. Open the Property Inspector and set the "MaxColumns" property to 3 and the "Rows" property to4. The visual editor immediately changes the input form layout accordingly

토= EmployeeSearch.jspx - Structure			
📌 🗟			
Warnings (2) Warnings (2) Warnings (2) Substrate the second state of the second st	 Panel Form La Panel Form La P Common Id: Rendered: MaxColumns: Rows: FieldWidth: LabelWidth: LabelWidth: LabelAlignment: 	yout - 4 - Property Inspector Pfil Cdefault> (true) 3 4 Cdefault>	
	Appearance		
	🗄 Style		

15. In the Data Controls panel, select the "Return" collection under the findEmployee(Object) | Return node. The collection object describes the structure or the Web Service response to display in the table.

WsSampleBaseDC Image: IndEmployee_parameters Image: IndEmployee(Object) Image: IndE
Recently Opened Files <<<<

16. Drop the "Return" collection as a table to the second facet of the af:panelSplitter component

17. Optionally, re-order the attributes in the table display. Ignore row selection, filtering and sorting for now.

18. Ok the dialog

🖮 Edit Table Columns		×
Enable ADF Behavior:	ow <u>S</u> election	Sorting
Columns:		🕂 💥 Group Ungroup
Display Label	Value Binding	Component To Use
🚥 <default></default>	💻 departmentId	ADF Output Text
<pre>xxx <default></default></pre>	📼 email	ADF Output Text
🚥 <default></default>	📼 employeeId	ADF Output Text
🚾 <default></default>	📼 firstName	ADF Output Text
Intersection of the second	📼 hireDate	ADF Output Text
🚥 <default></default>	📼 jobId	ADF Output Text
🚥 <default></default>	lastname	ADF Output Text
🚥 <default></default>	📼 managerId	ADF Output Text
<pre>xxi <default></default></pre>	phoneNumber	ADF Output Text
<pre>xxi <default></default></pre>	📼 salary	ADF Output Text
Help		Cancel

19. In the "Edit Action Binding" dialog there is nothing for you to do. Note the "Value" field for the "arg0" argument of the findEmployee(Object) method, which points to the variable iterator created when building the search form.

20. Close the dialog by clicking Ok.

💩 Edit Action Bir	ding		x
Select a data colle on the data object	ction and the action yo s of the selected colle	you want your control to initiate. The control initiates the ac ection.	tion.
Bata <u>C</u> ollection: 	BaseDC		
Select an Iterator:		eDCIterator Vew	
Operation:	findEmployee(Object	ect) 💌	
	Apply to all iter	rators in page defintion	
Parameters :			
Name	Туре	Value Option	
arg0	java.lang.Object	t \${bindings.arg0Iterator.currentRv	•
Help		OK Cancel	

21. To avoid full page refreshes when searching for employees, you need to configure Partial Page Refresh (PPR) behavior. For this, select the "Submit" button of the search form and set its PartialSubmit property to "true".

EmployeeSeard	ch.jspx												Ħ	Comp
🔹 Show 🕇 Fu	ull Screen Size	🕶 💽 🛛	2		▼Default		▼ None	- 🖏	🗞 🖉	P 🖪	ΙŪ	計~		1-(
Department Id	#{departme	ntld.inputVa	lue}		Hire Date	#{	.hireDate.inpi	utValue}		2	Phone	7	▶ ▼]	∙ly Ca IDE Ci
Mail	#{email.inpu	itValue}			Job Id	#{	.jobld.inputVa	ilue}			Salary		.	<u>d</u> 4
Employee Id	#{employee	ld.inputValu	e}		Last Name	#{	.lastname.inp	utValue]	}				÷	ا چھ ا
First Name	#{firstName	inputValue)			Manager	#{	.managerld.in	putValu	e}				<u>.</u>	.
Submit				5	Button - Si	ıbm	it - Propert	y Inspe	ctor					
				R	1 📌 🔮	1	/ 🔓 🍈	part				J	Ŷ) 🥐
			_		Pordor Col	or.								. 1
Department Id	Mail	in.	Emplo		Border Coll								ľ	
#{department	id} #{ema	iil}	#\c		Border Sty	le:							<u> </u>	
#{department	ld} #{ema	ii}	#{e		Border Wid	ith:			-	рх			• ~	·
					Outline Col	or:							• ~	·
					Outline Sty	le:							- -	11
					Outline Wid	dth:			-	Dx			Ī.	,
				1]
					Behavior									_
					Part <mark>i</mark> alSubmit	t: [<default> (f</default>	alse)					•	~
					PartialTrigge	rs:	<default> (fa</default>	alse)					,	~
					Nieskied.		false						_	
						1	a a c	42						

22. Select the table for the result set and search for its "PartialTriggers" property. The PartialTriggers property allows an ADF Faces component to listen for a change on another component (for example a button press).

23. Click the "arrow down" icon at the end of the PartialTriggers property and select the Edit option in the opened context menu. Oracle JDeveloper supports you in referencing the UI component to listen for, avoiding errors that may occur with components contained in naming containers.

Submit			Table - t1 - Property Ins	pector		
			🖳 📌 🖹 🥒 🖳	🎁 part	40	
Department Id	Mail	Emple	DisplayRow:	<default> (first)</default>	•	~ ^
#{departmentId}	#{email}	#{e	DisplayRowKey:			~
#{departmentId}	#{email}	#{e	EditingMode:	<default> (editAll)</default>	-	×
#1departmentidy	#\cinally	#1c	ContextMenuSelect:	<default> (true)</default>	•	×
			Immediate:	<default> (false)</default>	•	×
			PartialTriggers:			<u>~ _</u>
			RefreshCondition:			PartialTriggers X
			RangeSize:	25		Expression Builder
			RenderHint:	<default> (immediate)</default>	•	Reset to Default
			FilterModel:			Property Help
jsp:root ▼ } f:view ▼	af:document#d1	→ af:fo	rm#f1 → af:panelsplitter#	ips1 → > fifacet → > afitable#t1 →	2~	the IDs of the components that should trigger a partial update. This component will listen on the trigger components. If one of the trigger components receives an
Design Source Bindir	ngs Preview Histo	ory <		2		ouont that will cause it to undate

Edit Property: PartialTriggers Shuttle items from Available to Select shuttled, it requires an Id or an ance relative and absolute path reference	ted to define the list of scoped ids for this co stor naming container requires an Id. Note t s depending on their relationship to the curr	omponent. If an item cannot be that scoped ids use a combination of ent component.
Available: Availa	selected: xt - it5 xt - it9 xt - it4 ite - id1 xt - it8 xt - it2 xt - it1 xt - it3 xt - it6 iooter) mandButton - cb1	
		OK Cancel

24. In the "Edit Property" dialog, find the submit button. The submit button is not shown by its label "submit" in the editor, but its component Id. You are able to find the button by its location, which is the first PanelSplitter facet, Panel Form Layout footer facet

25. Drag the findEmployee(Object) method from the Data Controls panel to the JSF page



26. Drop the findEmployee(Object) method onto the submit button. The submit button so far would only submit the search form to the server, but would not yet invoke the findEmployee method to query the list of employees from the Web Service. Dropping the findEmployee(Object) method onto the submit button creates a method binding entry in the PageDef file and also configures the submit button ActionListener property to reference this binding.



27. In the "Component Rebinding" dialog, select the "text" property and the "disabled" property as you want to keep the current configuration for these.

28. Ok the dialog



29. To ensure all search criteria is right aligned in the input text fields, a bit of CSS is required. Keep the CTRL-Key pressed and click on all components contained in the Panel Form Layout. The best place to do this is the Structure Window

30. Open the Property Inspector, navigate to the "ContentStyle" property and add

text-align:right;

The "ContentStyle" property ensures the CSS is directly applied to value area of the input components.

nelSplitter - vertical	Multiple - Property Ins	pector	Firethlong
	💽 I 🖈 🖹 I 🥒 🌘	Find Vî) ?
umia ar:panelFormLayout - 4 imi∰ af:inputText - #{bindings.departmentId.hints.label}	- ClientComponent:	<default> (false)</default>	~ ^
····· → af:inputText - #{bindings.email.hints.label}	Columns:	#{bindings.managerId.hints.displayWidth}	~ _
in the string of	ContentStyle:	text-align:right;	~
af:inputText - #{bindings.jobId.hints.label}	Converter:	<default> ↓ ▼</default>	~ 1
af:inputText - #{bindings.lastname.hints.labe}	Customization allowed:	<default></default>	~ L
af:inputText - #{bindings.phoneNumber.hints.labe/	Customization allowed by:		~
af:inputText - #{bindings.salary.hints.label}	Disabled:	<default> (false)</default>	~
ar:inputtext - #{bindings.phonendumber.nints.label}	HelpTopicId:		ř
🗄 🗠 🔄 Panel Form Layout facets	• IC:		č
secona		<pre><derauit>(raise)</derauit></pre>	× •//

31. Select the EmployeeSearch.jspx page and choose "Run" from the context menu

EmployeeSearch iso		#5	denartmentId\
DcModel		<u>O</u> pen	
		Exclude Project Content	
	×	<u>D</u> elete	
	5	<u>R</u> eformat	Ctrl+Alt-L
	2	Go to Page Definition	
		<u>F</u> ind Usages	Ctrl+Alt-U
		Refa <u>c</u> tor	•
-		<u>M</u> ake	Ctrl+Shift-F9
n Resources		Rebuild	Alt+Shift-F9
itrols	_	_	
Opened Files	\triangleright	Run	Ctrl-F11
	۲	De <u>b</u> ugs	
eSearch.jspx - Structure		Compare Wit <u>h</u>	•
		Replace <u>W</u> ith	•
nings (4) oot		Crea <u>t</u> e Portlet Entry	

32. Note the search field and table header labels, which are read from the Data Control UI Hints you created before.

33. Move the mouse over a search field and wait for the tool tip to show – which is also read from the DataControl configuration

🥖 🏉 http://127.0).0.1:7101/WsSamp	le-ViewController-		à				
Department Id		50	Hire Date	10	Phone			
Mail			Job Id	ST_	Salary			
Employee Id		L	ast Name	Ki	Phone			
First Name		M	anager Id		Salary			
	h. d							
Department Id	Mail	Employee Id	First Name	Hire Date	Job Id	Last Name	Manager	
Department Id	Mail IMIKKILI	Employee Id	First Name Irene	Hire Date 9/28/1998	Job Id ST_CLERK	Last Name Mikkilineni	Manager 120	

34. Type values into the search form field, as shown above, and press "Submit". The table should refresh with the data queried from the WS

Adding Model Validation

An advantage of using the WS Data Control is that it allows you to define validation rules, like range validation or pattern validation, like mail address verification. So let's add validation to the project. What about a pattern validation that ensures the entered departmentId to search for is between 10 and 999 ?

1. In the WsDcModel project, select the "arg0" entry contained in the WsSampleBaceDC.findEmployees.params package

PApplication D. 3 R. R. R.	xyzarg0.xml	
🕒 WsSampleBase 🗸 🕞 🗸	Attributes	Attributes
	Operations	Name Type Defa SVB departmentId java.lang.Long SVB email java.lang.String SVB employeeId java.lang.String SVB firstName java.lang.String SVB hireDate java.lang.String SVB jobId java.lang.String
midEnployee_Operations.xml midEmployee.xml midEmployee.xml midEmployee.xml midEmployee.xml midEmployee.params midEmployee.params midEmployee.xml midEmployee.params midEmployee.xml m		Iastname java.lang.String ImanagerId java.lang.Long ImanagerId java.lang.String ImanagerId java.lang.Long
Application Resources Data Controls Recently Opened Files		

2. Double click the "arg0" entry to open the attribute editor.

📥 Edit Attribute		X
🏽 🎁 Search	Validation	
Generic Attribute Control Hints	Hint: Select an attribute and click the New button to apply a new validation rule.	
Validation Custom Properties	Declared Validation Rules:	
Help		Cancel

3. Select the departmentId entry and click the edit icon (pencil icon)

4. In the "Edit Attribute" dialog, select the "Validation" entry and click the "New" button

📥 Add Valida	tion Rule for: departmentId	x
Define the Va	lidation you want to perform with this rule and configure the Validation Failure response	
<u>R</u> ule Type:	Compare	-
	Compare	
Rule Definit	Length List	
<u>A</u> ttribute:	Range	
Operator:	Regular Expression	
Compare V	Script Expression	
Enter Lite	ral Value	
		ר
Hint: Enter	a valid value for the selected attribute's type.	
Help	OK Cano	-

5. Select "Regular Expression" as the "Rule Type"

Define the Validation you want to perform with this rule and configure the Validation Failure response.									
Rule Type: Regular Expression									
Rule Definition Failure Handling									
Attribute: departmentId									
Operator: Matches									
Select a predefined expression and click Add to insert the definiton below									
Predefined Expressions:									
Enter Regular Expression									
^[+]?[1-9]\d{1,2}(\.\d{1,2})?%?\$									
Expression Qualifiers									
Case Insensitive Multiline Canon Eq									
DotAll Unicode Case									
Hint: Enter a valid regular expression.									
Help Cancel									

6. In the "Enter Regular Expression", add

^[+]?[1-9]\d{1,2}(\.\d{1,2})?%?\$

Note: You can copy and paste this String from above

7. Select the "Failure Handling" tab Here we specify an error message, which also can be translated into different languages

🕹 Add Validation Rule for: departmentId	
Define the Validation you want to perform with this rule and configure the Validation Failure response.	
Rule Type: Regular Expression	
Rule Definition Failure Handling	
Validation Failure Severity ④ Error 〇 Informational Warning	
Failure Message	
Enter text for the translatable validation failure messages.	
Message Text:	
Department ID value {departmentId} is incorrect. Value must be higher than 10 but less than 1	
Token Message Expressions:	
Message Token Expression	
departmentId departmentId	
Help OK Cancel	

8. Add a message string like above saying that the provided value must be between 10 and 999. Make sure you add {departmentId} somewhere in your text.

9. The {departmentId} flag becomes a variable that allows you to add information from the attribute into your message. In the example we want to tell the user that the value he/she entered is not valid. For this add "departmentId" to the Expression field.

10. To see why and how this works, go to the ViewController project and select the departmentId input text component "f:validator" tag in the Structure Window and open the Property Inspector. Note that the "Binding" points to the ADF binding's department Id attribute binding, and here the "validator" attribute

🔠 af:panelFormLayout - 4						
🚊 🖞 af:inputText - #{bindir	ngs.departmentI	d.hints.label}	#{0	departmentId.	#{email.label}	#
f:validator	ኛ Validator -	Property Ins	pector	l		
🗄 🗟 Input Text facet	强 i 📌 📴	I 🥒 🤇 (🎁	Find		ふう ふう ふう しょう しょう しょう しょう しょう しょう しょう しょう しょう しょ	?
⊕…∰ af:inputText - #{bind						
⊜…∰ af:inputText - #{bind	ValidatorId:	<default></default>			-	~
🗝 🤴 f:validator	Binding:	#{bindings.de	partmentId.	validator}		\sim
123 af:convertNumbe		L				
🗄 🗟 Input Text facet						
🗈 🖞 af:inputText - #{bind						
🗄 🖽 af:inputDate - #{bind						
⊕ ∰ af:inputText - #{bind				2		
⊕ ∰ af:inputText - #{bind						
⊕ ∰af:inputText - #{bind						
⊕ ∰ af:inputText - #{bind						
⊕ ∰ af:inputText - #{bind						
🗄 🗠 🔜 Panel Form Layout fa						
second						
af:table - t1			_			_//.

11. Rerun the search page and type 1200 into the departmentId field. Press submit and see the error message thrown by the binding layer.

🖉 http://127.0.0	0.1:7101/WsSample-V	iewController			
Department Id		1200 Hire	Date	1	Phone
Mail Employee Id			Error: Departme must be higher t	nt ID value 1200 is han 10 but less tha	incorrect. Value an 1000
First Name		L	han 10 but less the	an 1000	aiue must be nigner
Submit					
Department Id	Mail	Employee Id	First Name	Hire Date	Job Id
10	JWHALEN	200	Jennifer	9/17/1987	AD_ASST
20	MHARTSTE	201	Michael	2/17/1996	MK_MAN
22	DE AM			a lum lunna	

Enable Table Filtering and Sorting

As a last exercise in this hands-on, you add table filter and sorting functionality to the result table.

Data Controls	1 2						
Recently Opened Files	Departm	ent Id	Mail	Employe	eld	First Name	H
644	#{depa	artmentId}	#{email	} #{empl	oyeeld}	#{firstName}	#
	#{depa	artmentId}	#{email	} #{empl	oyeeld}	#{firstName}	#
EmployeeSearch.jspx - Structure	#{depa	artmentId}	#{email	} #{empl	oyeeld}	#{firstName}	#
EmployeeSearch.jspx - Structure Marnings (4) Gamma Signary Structure Marnings (4) Gamma Signary Structure Gamma Signary Structure	\$#{deps	Table Table Comm Gamm Gamm Gamm Gamm Gamm Gamm Gamm G	#{email		oyeeld}	#{firstName}	*
	Second						- 1 li
i⊒≣ af:table - t1	jsp:root	AllDetai	lsEnabled:	<default> (false)</default>		-	~
	Design	🛙 Value:		#{bindings.Return.col	lectionModel}		~
⊕— ⊟ af:column - #{bindings.Re ⊕— ⊟ af:column - #{bindings.Re ⊕— ⊟ af:column - #{bindings.Re	Runnir	 Colum ★ Appea	ns rance				_
i⊞… i af:column - #{bindings.Re i ⊡… i af:column - #{bindings.Re	[06:52: [06:52:	🗆 Style					

1. Select the table component in the Structure Window and open the Property Inspector

2. In the Property Inspector, click the "pencil" icon to launch the component configuration

3. In the "Edit Table Column" dialog, check the "Sorting" and "Filtering" option. The "Filtering" option adds a search binding in the PageDef configuration file and references it from the table's QueryListsner property

📥 Edit Table Columns		X
Enable ADF Behavior: Row	Selection	Sorting
Columns:		🕂 💥 Group Ungroup
Display Label	Value Binding	Component To Use
🚥 <default></default>	💻 departmentId	ADF Output Text
<pre>xxa <default></default></pre>	📼 email	ADF Output Text
<pre>ssa <default></default></pre>	📼 employeeId	ADF Output Text
🚥 <default></default>	📼 firstName	ADF Output Text
<pre>ssa <default></default></pre>	📼 hireDate	ADF Output Text
<pre>ssa <default></default></pre>	📼 jobId	ADF Output Text
<pre>xxa <default></default></pre>	📼 lastname	ADF Output Text
<pre>xxa <default></default></pre>	📼 managerId	ADF Output Text
<pre>xxa <default></default></pre>	phoneNumber	ADF Output Text
<pre>sea <default></default></pre>	📼 salary	ADF Output Text
		'₹
Help		Cancel

4. Click OK and run the search page from within JDeveloper

🏉 http://127.0.0	0.1:7101/WsSample	e-ViewContro	ller				🖄 🔻 6	3 -
Department Id		50	Hire Date		E E	hone		
Mail			Job Id		9	Salary		
Employee Id			Last Name					
First Name			Manager Id					
Submit			-					
					20	ST_CLERK		
Department Id	Mail	Employe	e Id	First Name	Hire Date	Job Id	Last Name	Man
50	JNAYER.	125		Julia	7/16/1997	ST_CLERK	Nayer	120
50	IMIKKILI	126		Irene	9/28/1998	ST_CLERK	Mikkilineni	120
50	JLANDRY	127		James	1/14/1999	ST_CLERK	Landry	120
50	SMARKLE	128		Steven	3/8/2000	ST_CLERK	Markle	120
50	LBISSOT	129		Laura	8/20/1997	ST_CLERK	Bissot	121
50	MATKINSO	130		Mozhe	10/30/1997	ST_CLERK	Atkinson	121
50	JAMRLOW	131		James	2/16/1997	ST_CLERK	Marlow	121
50	TJOLSON	132		τJ	4/10/1999	ST_CLERK	Olson	121
50	JMALLIN	133		Jason	6/14/1996	ST_CLERK	Mallin	122
50	MROGERS	134		Michael	8/26/1998	ST_CLERK	Rogers	122
50	KGEE	135		Ki	12/12/1999	ST_CLERK	Gee	122
50	HPHILTAN	136		Hazel	2/6/2000	ST_CLERK	Philtanker	122
							1 1 1 1	100
50	RLADWIG	137		Renske	7/14/1995	ST_CLERK	Ladwig	125

- 5. In the search form, enter 50 for the departmentId and press the Submit button
- 6. Type ST_CLERK into the JOB_ID filter of the table.
- 7. Hit the ENTER key to filter the table
- 8. Click the Sort icon in the "MAIL" column header to sort the queried and filtered result set

Department Id		50	Hire Date		20	P	hone	
Mail			Job Id			s	alary	
Employee Id			Last Name			1		
First Name			Manager Id			1		
First Name			Manager 10					
						20	ST_CLERK	
Department Id	Mail 🛆	Employee	e Id	First Name	Hire Date	20	ST_CLERK Job Id	
Department Id 50	Mail JNAYER	Employee	e Id	First Name Julia	Hire Date 7/16/1997	2	ST_CLERK Job Id ST_CLERK]] 1
Department Id 50 50	Mail A	Employee	e Id nding	First Name Julia Irene	Hire Date 7/16/1997 9/28/1998	20	ST_CLERK Job Id ST_CLERK ST_CLERK	
Department Id 50 50 50	Mail	Employee 125 Sort Descer 127	e Id	First Name Julia Irene James	Hire Date 7/16/1997 9/28/1998 1/14/1999	20	ST_CLERK Job Id ST_CLERK ST_CLERK ST_CLERK] 1 1 1 1
Department Id 50 50 50 50	Mail JNAYER IMIKKILI JLANDRY SMARKLE	Employee 125 Sort Descer 127 128	e Id	First Name Julia Irene James Steven	Hire Date 7/16/1997 9/28/1998 1/14/1999 3/8/2000	1	ST_CLERK Job Id ST_CLERK ST_CLERK ST_CLERK ST_CLERK	
Department Id 50 50 50 50 50	Mail JNAYER IMIKKILI JLANDRY SMARKLE LBISSOT	Employee 125 Sort Descen 127 128 129	e Id nding	First Name Julia Irene James Steven Laura	Hire Date 7/16/1997 9/28/1998 1/14/1999 3/8/2000 8/20/1997	1	ST_CLERK Job Id ST_CLERK ST_CLERK ST_CLERK ST_CLERK ST_CLERK	
Department Id 50 50 50 50 50 50 50	Mail JNAYER IMIKKILI JLANDRY SMARKLE LBISSOT MATKINSO	Employee 125 Sort Descen 127 128 129 130	e Id	First Name Julia Irene James Steven Laura Mozhe	Hire Date 7/16/1997 9/28/1998 1/14/1999 3/8/2000 8/20/1997 10/30/1997	b	ST_CLERK Job Id ST_CLERK ST_CLERK ST_CLERK ST_CLERK ST_CLERK ST_CLERK] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
Department Id 50 50 50 50 50 50 50 50	Mail JNAYER IMIKKILI JLANDRY SMARKLE LBISSOT MATKINSO JAMRLOW	Employee 125 Sort Descer 127 128 129 130 131	e Id	First Name Julia Irene James Steven Laura Mozhe James	Hire Date 7/16/1997 9/28/1998 1/14/1999 3/8/2000 8/20/1997 10/30/1997 2/16/1997	1	ST_CLERK Job Id ST_CLERK ST_CLERK ST_CLERK ST_CLERK ST_CLERK ST_CLERK ST_CLERK	
Department Id 50 50 50 50 50 50 50 50 50	Mail JNAYER IMIKKILI JLANDRY SMARKLE LBISSOT MATKINSO JAMRLOW TJOLSON	Employee 125 Sort Descen 127 128 129 130 131 132	e Id nding	First Name Julia Irene James Steven Laura Mozhe James TJ	Hire Date 7/16/1997 9/28/1998 1/14/1999 3/8/2000 8/20/1997 10/30/1997 2/16/1997 4/10/1999	b	ST_CLERK Job Id ST_CLERK ST_CLERK ST_CLERK ST_CLERK ST_CLERK ST_CLERK ST_CLERK ST_CLERK	
Department Id 50 50 50 50 50 50 50 50 50 50 50	Mail JNAYER IMIKKILI JLANDRY SMARKLE LBISSOT MATKINSO JAMRLOW TJOLSON JMALLIN	Employee 125 Sort Descen 127 128 129 130 131 132 133	e Id nding	First Name Julia Irene James Steven Laura Mozhe James TJ Jason	Hire Date 7/16/1997 9/28/1998 1/14/1999 3/8/2000 8/20/1997 10/30/1997 2/16/1997 4/10/1999 6/14/1996	- B	ST_CLERK Job Id ST_CLERK ST_CLERK ST_CLERK ST_CLERK ST_CLERK ST_CLERK ST_CLERK ST_CLERK ST_CLERK	

Downloads

The Oracle JDevloper starter workspace can be downloaded in a ZIP file as sample #73 from the ADF Code Corner website at

http://www.oracle.com/technetwork/developer-tools/adf/learnmore/index-101235.html

RELATED DOCOMENTATION