



# Contents at a Glance

## PART I Core PeopleTools Concepts

1	Application Classes .....	3
2	The File Attachment API .....	41
3	Approval Workflow Engine .....	91
4	Pagelet Wizard .....	141

## PART II Extend the User Interface

5	Understanding and Creating iScripts .....	187
6	JavaScript for the PeopleSoft Developer .....	229
7	AJAX and PeopleSoft .....	263
8	Creating Custom Tools .....	303

## PART III Java

9	Extending PeopleCode with Java .....	337
10	A Logging Framework for PeopleCode .....	387
11	Writing Your Own Java .....	415
12	Creating Real-Time Integrations .....	463
13	Java on the Web Server .....	505
14	Creating Mobile Applications for PeopleSoft .....	519

**PART IV**  
**Best Practices**

<b>15</b>	<b>Test-Driven Development</b> .....	<b>561</b>
<b>16</b>	<b>PeopleCode Language Arts</b> .....	<b>579</b>
	<b>Index</b> .....	<b>595</b>

# Contents

Acknowledgments .....	xvii
Introduction .....	xix

## PART I

### Core PeopleTools Concepts

<b>1</b> Application Classes .....	3
Our First Application Class .....	4
Creating an Application Package and Class .....	4
Coding the Application Class .....	5
Testing the Application Class Code .....	6
Expanding the Application Class .....	11
Inheritance .....	13
Features of Application Classes .....	17
Dynamic Execution .....	17
Construction .....	17
Stateful Objects .....	17
Access Control .....	19
Putting It All Together: The Logging Framework Example .....	21
Log Levels .....	22
The Logger Interface .....	25
The Logger Classes .....	25
Testing the Framework .....	31
Dynamic Logger Configuration .....	32
Factory Test Program .....	36
Misuses of Application Classes .....	38
Runtime Context-Sensitive Variables .....	38
Indiscriminate Usage .....	38
Conclusion .....	39
Notes .....	39

## X PeopleSoft PeopleTools Tips & Techniques

<b>2</b>	<b>The File Attachment API</b>	<b>41</b>
	Adding Attachments to Transactions	42
	Investigating the Target Transaction	42
	Creating the Attachment Storage Record	49
	Adding the FILE_ATTACH_SBR Subrecord	53
	Adding Attachment Fields and Buttons to the Transaction Page	53
	Writing PeopleCode for the Attachment Buttons	59
	Customizing File Attachment Behavior	65
	Moving to Level 1	77
	Modifying the Page	77
	Adding the PeopleCode	81
	Adding Multiple Attachments per Transaction	85
	Processing Attachments	85
	Accessing Attachments	86
	Storing Attachments	87
	Implementing File Attachment Validation	87
	Filename Validation	87
	File Contents Validation	89
	Conclusion	89
<b>3</b>	<b>Approval Workflow Engine</b>	<b>91</b>
	Workflow-Enabling Transactions	92
	Creating Supporting Definitions	93
	Configuring the AWE Metadata	102
	Modifying the Transaction	108
	Testing the Approval	126
	Providing Custom Descriptions for the Approval Status Monitor	129
	Allowing Ad Hoc Access	132
	Creating an Event Handler Iterator	136
	Web Service-Enabling Approvals	138
	Conclusion	140
<b>4</b>	<b>Pagelet Wizard</b>	<b>141</b>
	Pagelets Defined	142
	Creating a Pagelet	143
	Components of a Pagelet Wizard Pagelet	148
	Pagelet Data Types	148
	Setup for the Custom Data Type Example	148
	Coding the Custom Data Type	149
	Registering the Data Type	168
	Creating a Test Pagelet	168
	Pagelet Transformers	177
	XSL Templates	180
	Display Formats	180
	Conclusion	183
	Notes	183

## PART II

### Extend the User Interface

<b>5</b>	Understanding and Creating iScripts .....	187
	iScripts Defined .....	188
	Our First iScript .....	189
	Coding the iScript .....	189
	Testing the iScript .....	190
	Modifying the iScript .....	191
	A Bookmarklet to Call an iScript .....	192
	Writing the SetTraceSQL iScript .....	193
	Creating a Bookmarklet .....	194
	Desktop Integration .....	197
	Creating an iScript to Serve Calendar Content .....	198
	Building a Parameter Cache .....	200
	Modifying the Transaction .....	203
	Serving File Attachments .....	209
	iScripts as Data Sources .....	210
	Flex Requirements .....	211
	Say Hello to Flex .....	211
	Direct Reports DisplayShelf .....	220
	Conclusion .....	227
	Notes .....	227
<b>6</b>	JavaScript for the PeopleSoft Developer .....	229
	A Static JavaScript Example .....	230
	A Dynamic JavaScript Example .....	233
	Creating the Derived/Work Record for Dynamic HTML .....	234
	Adding PeopleCode for the HTML Area .....	234
	Creating an HTML Definition .....	235
	Inspecting PeopleSoft's User Interface with Firebug .....	236
	Using Firebug's Console .....	236
	Using Firebug to Enhance the Trace Bookmarklet .....	237
	Styling an Element .....	239
	JavaScript Libraries .....	240
	Serving JavaScript Libraries .....	241
	Using jQuery .....	243
	Making Global User Interface Changes .....	245
	Identifying Common Definitions .....	245
	Minimizing the Impact .....	247
	Coding the Solution .....	247
	Using jQuery Plug-ins .....	254
	Thickbox .....	254
	WEBLIB_APT_JSL iScript Code .....	260
	Performance Issues .....	261
	Conclusion .....	261
	Notes .....	261

<b>7</b>	<b>AJAX and PeopleSoft</b>	<b>263</b>
	Hello AJAX	264
	Creating the AJAX Request Handler	264
	Ajaxifying a Page	264
	Adding Animation	266
	Ajaxifying the Direct Reports DisplayShelf	268
	Modifying the Flex Source	269
	Modifying the Direct Reports Service	270
	Creating a New HTML AJAX Service	271
	Modifying the Container Page	273
	A Configurable User Interface	275
	Using a Metadata Repository	275
	Modifying the Bootstrap Code	285
	Testing the Custom Scripts Component	291
	Highlight Active Field Revisited	291
	Changing Search Operators	293
	Fiddler	298
	Conclusion	301
	Notes	302
<b>8</b>	<b>Creating Custom Tools</b>	<b>303</b>
	The Toolbar Button Metadata Repository	304
	Setting Up the Repository Tables	304
	Creating the Toolbar Maintenance Page	308
	Defining the Toolbar's HTML	311
	Attaching the Toolbar to Pages	312
	Defining a Custom Script for the Toolbar	312
	Adding the Trace Toolbar Button	314
	Modifying the Bootstrap Code	316
	Separating Common Code from Bootstrap Code	316
	Adding New URL-Generation Functions	317
	Writing the New Common Code	318
	Launching Another Component	324
	Creating an iScript to Get CREF Information	324
	Adding the Edit CREF Toolbar Button	325
	Viewing Query Results	328
	Creating a Query to Get a Page's Permission Lists	328
	Adding the Query Toolbar Button	329
	Leaving the Portal	333
	Conclusion	334

**PART III****Java**

<b>9</b>	Extending PeopleCode with Java	337
	Java Overview	338
	Why Java?	338
	Why Not C++ or .NET or ...?	339
	Java and PeopleCode 101	339
	Java Strings	339
	Java Arrays	347
	Java Collections	349
	Writing a Meta-HTML Processor	350
	Implementing %Image	350
	Implementing %JavaScript	353
	Implementing %GenerateQueryContentURL	356
	Complete Code for the Meta-HTML Processor	361
	Using Third-Party Libraries	364
	Apache Commons	365
	Apache Velocity	368
	Using JSON	373
	Conclusion	385
	Notes	385
<b>10</b>	A Logging Framework for PeopleCode	387
	Investigating Problems	388
	Delivered Logging Tools	388
	The log4j Java Logging Framework	389
	Hello log4j	389
	Tracing log4j	392
	Configuring log4j	393
	Improving Logging Performance	397
	Avoiding Logger Reconfiguration	397
	Using log4j in the Process Scheduler	398
	An Integrated Logging Framework	398
	Creating the Level Class	399
	Creating the Logger Class	400
	The LogManager Class	403
	log4j Metadata	410
	Testing APT_LOG4J	411
	Conclusion	413
	Notes	413

<b>11</b>	Writing Your Own Java	415
	Your Java Build Environment	416
	Your First Java Class	416
	Creating the Source Files	416
	Deploying Java	420
	Creating the Test Program	426
	Using PeopleCode Objects in Java	426
	Configuring Your Development Environment	426
	Using PeopleCode System Variables	427
	Accessing Data	429
	PeopleSoft Database log4j Appender	436
	Creating the PL/SQL Autonomous Transaction	437
	Writing the Java	441
	Testing the Appender	448
	Static Configuration	449
	PeopleSoft Database Velocity Template Data Source	450
	Creating the Template Metadata Repository	450
	Creating the Velocity Repository Java Class	453
	Testing the PSDBResourceLoader	460
	Multithreading	461
	Conclusion	461
	Notes	462
<b>12</b>	Creating Real-Time Integrations	463
	Integration Technologies	464
	Setting Up for Database Integration	464
	Creating a Custom JDBC Target Connector	466
	Creating the JDBCTargetConnector Class	466
	Predeployment Testing	482
	Deploying the Connector	492
	Configuring Integrations	492
	Configuring the Gateway	493
	Creating a Node	494
	Transforming Messages	495
	Creating a Routing	499
	Testing the Integrated Connector	501
	Troubleshooting Custom Connectors	502
	Conclusion	504
	Notes	504

<b>13</b>	Java on the Web Server	505
	Extending the PeopleSoft Web Server with JSP	506
	Using Servlet Filters	507
	Investigating iScript Caching Behavior	508
	Creating an HTTP Header Servlet Filter	509
	Testing the Servlet Filter	513
	Deploying the Servlet Filter	515
	Conclusion	516
	Notes	517
<b>14</b>	Creating Mobile Applications for PeopleSoft	519
	Providing Web Services	520
	Enabling a Component Interface as a Web Service	520
	Testing the WSDL URL	528
	Going Mobile with JDeveloper	528
	Creating a Fusion Web Application	529
	Creating the Data Control	532
	Creating the View	536
	Designing the Search Page	540
	Testing the Search Page	553
	Shortening the Application's URL	555
	Requiring Authentication	556
	Conclusion	557
	Notes	558

## PART IV

### Best Practices

<b>15</b>	Test-Driven Development	561
	Introduction to Test-Driven Development	562
	The TDD Approach	562
	Some TDD Lingo	563
	A TDD Framework	564
	Test Driving the Meta-HTML Processor	564
	Writing a Test	564
	Running the Test	566
	Making the Test Pass	569
	Running the Test Again	569
	Refactoring	570
	Repeating the Cycle	573
	Conclusion	578
	Notes	578

<b>16</b>	PeopleCode Language Arts .....	579
	Composition over Inheritance .....	580
	Façades .....	581
	Factories .....	582
	Inversion of Control .....	583
	Enumerated Types .....	591
	Language Diversity .....	593
	Notes .....	594
	Index .....	595