

Extensis™
Portfolio 6

PortWeb User Guide
for Macintosh® and Windows®



Extensis, Inc.

1800 SW First Avenue, Suite 500
Portland, OR 97201

Phone: (503) 274-2020

Fax: (503) 274-0530

E-mail: info@extensis.com

Internet: <http://www.extensis.com>

Customer Service

Phone: 800-796-9798

E-mail: sales@extensis.com

Technical Support

Web: <http://www.extensis.com/support/>

Phone: 503-274-7030

Monday–Friday

8am–5pm Pacific

Extensis Europe

First Floor

Century House

The Lakes, Northampton NN4 7SJ

United Kingdom

Telephone: +44 (0)16 04 636300

Fax: +44 (0)16 04 636366

E-mail: info@extensis.co.uk

© 2002 Extensis, Inc. This manual and the software described in it are copyrighted with all rights reserved. Under the copyright laws, neither this manual nor the software may be copied, in whole or part, without the written consent of Extensis, Inc., except in the normal use of the software or to make a backup copy of the software. This exception does not allow copies to be made for others.

Extensis, Inc., Extensis, the Extensis logo, and Extensis Library are trademarks of Extensis, Inc. PostScript is a trademark of Adobe Systems, Incorporated. Apple, Macintosh, PowerPC, AppleScript, FontSync, and QuickDraw are registered trademarks of Apple Computer, Inc. FontBook is a trademark of 2000 Lemke Software. Netscape and Navigator are registered trademarks of Netscape Communications. Microsoft and Internet Explorer are registered trademarks of Microsoft Corporation. Symantec and Suitcase are trademarks of Symantec Corporation. Quark and QuarkXPress are registered trademarks and XTensions is a trademark of Quark, Inc. All other trademarks are the property of their respective owners.

Extensis, Inc. warrants the disks on which the software is recorded to be free from defects in materials and faulty workmanship under normal use for a period of ninety (90) days from the original date of purchase. If a defect occurs during the 90-day period, you may return the disks to Extensis, Inc. for a free replacement provided that you have previously registered your product with Extensis, Inc. This warranty is limited to replacement and shall not encompass any other damages, including but not limited to loss of profit, and special, incidental, or other similar claims. This software is provided on an "as is" basis. Except for the express warranty set forth above, Extensis, Inc. makes no other warranties, either express or implied, regarding the enclosed software's quality, performance, merchantability, or fitness for a particular purpose.

Quark, Inc. makes no warranties, either express or implied, regarding the enclosed computer software package, its merchantability or its fitness for any particular purpose. Quark, Inc. disclaims all warranties of including, but not limited to, distributors and retailers of Quark, Inc. software. Without limiting the foregoing, in no event shall Quark, Inc. be liable for any special, indirect, incidental or consequential damages in any way relating to the use or arising out of the use of the enclosed software. Quark, Inc.'s liability shall in no event exceed the total amount of the purchase price/license fee actually paid for the use of the enclosed software.

Introducing PortWeb

PortWeb 6.0, a web server plug-in, is an easy-to-use add-on to the Portfolio 6.0 asset management system that enables web browser-based users to browse Portfolio catalogs. PortWeb establishes a connection between a web server and either a Portfolio Server or a Portfolio catalog on disk to provide a truly dynamic link between web-based users and Portfolio catalogs, so they can quickly search up-to-date Portfolio information.

System Requirements

To install and use PortWeb 6.0, you will need the following:

Macintosh

- PPC Only
- Minimum OS: System 8.6
- Supported Web Servers:
- StarNine WebStar 3.0–4.x
- Apple Web Server (Part of Appleshare IP 6.3)

Windows

- Pentium 200
- Windows 98/2000/NT 4.0/XP
- Supported Web Servers
- MS Internet Information Server (IIS) (3.0–5.0)
- MS Personal Web Sharing Server (Win 98)
- MS Peer to Peer Web Sharing Services (NT,2000)

Required Knowledge

Setting up and administering PortWeb is not difficult, but it does require a good knowledge of HTML and experience with setting up and administering a Mac or Windows web server. The instructions in this User Guide assume that you are already comfortable building HTML pages, creating HTML forms and administering a web server.



For additional support in working with PortWeb, we recommend consulting the following web site, where you can find templates, sample sites, scripts and other helpful items:

<http://portweb.extensis.com>

Overview of PortWeb

PortWeb is a web server plug-in. A web server plug-in extends the functionality of the web server. Essentially, when a web server is launched, it looks to determine whether any plug-ins are installed. If plug-ins are found, they are loaded, so that the web server becomes aware of the new types of requests that can be handled, and which plug-in should receive the request.

PortWeb works by registering itself with the web server, so that the web server knows to pass certain types of requests on to PortWeb. The general process that occurs in a user's search through PortWeb is as follows:

- 1 The user loads a static HTML search page containing a search form.

- 2 The user enters search criteria and submits the form.
- 3 The web server passes the form data over to PortWeb.
- 4 PortWeb parses the data and performs the search in a specified Portfolio catalog.
- 5 If a specific HTML template has been specified, PortWeb parses the template.
- 6 PortWeb merges the search results with the template and returns the page back to the user's web browser (via the web server) for display. If no template was specified, PortWeb builds its own "generic" HTML results page, which contains thumbnails and filenames, but no links to previews or original images.

Web Server Authentication

In order to access the PortWeb Administration Module and begin configuring it, you must have *administrative access* to the web server. PortWeb will attempt to authenticate that you have such privileges before displaying the Administration module page. Most web servers have their own unique mechanism for validating administrators.

Some web servers do not have any authentication at all; in these situations, PortWeb will check to see whether you are accessing the administration page from the same machine (based on IP address) as the web server itself. If you attempt to access the Administration module from a remote machine on a web server that does not support authentication, the attempt will fail.

WebStar 3.0 and 4.0: If you access PortWeb administration through the usual URL (<http://yourserver/.PortWeb?adminshow>) or

through WebStar's administration page, PortWeb will allow any user in the Administration realm to read and change settings. You can also specify the realm for PortWeb administration (i.e. http://yourserver/pi_admin.PortWeb?adminshow), and PortWeb will allow users in that realm to access PortWeb administration.

(See the WebStar manual for more information on security, realms, and users.)

AppleShare IP: PortWeb requires that you access the Administration module from the machine running PortWeb.

Windows NT 4.0/Windows 2000 Server Internet Information Server and Windows NT 4.0/Windows 2000

Workstation Peer Web Services: With the "Basic Authentication" setting checked in the Service Properties dialog, the user will be prompted for a name and password. Any user who belongs to the "Administrators" group on the server computer will be properly authenticated and allowed to view and edit the PortWeb Administration module.

Personal Web Sharing Server for

Windows 98: Most versions of PWS do not support authentication. In these situations, you must access the Administration module from the machine running PortWeb.

Setup and Administration

Installing PortWeb

You can find instructions for quick and easy installation in the READ ME file that is included with PortWeb.

Before installing PortWeb, confirm that you have one of the supported web servers (listed on page 3) installed on your computer.

When you install PortWeb, the PortWeb plug-in should be copied to the appropriate folder location on your system so that it can be accessed by your web server software.

To use PortWeb, you will also need to install Portfolio 6 to create and maintain the catalogs that you will publish using PortWeb. Refer to the *Portfolio 6 User Guide* for instructions on how to install and use Portfolio.

Macintosh Memory Usage

As a plug-in, PortWeb does not allocate its own memory partition, but rather operates within the memory partition of the web server. Because PortWeb needs more memory than most plug-ins, you should increase the amount of memory that your web server allows the plug-in after you have installed PortWeb.

WebStar:

- 1 Make sure that WebStar is not running.
- 2 Select the WebStar application in the Finder.
- 3 Choose “Get Info” from the File menu.
- 4 Increase the “Preferred Size” by 3000 K.

Appleshare IP 6.x:

- 1 Open the Appleshare IP Web & File Admin application
- 2 Choose “Web & File Server Settings” from the Server menu.
- 3 Switch to the Web Plug-Ins pane.
- 4 Increase the “Maximum Plug-in Memory Usage” by 3000 K.
- 5 Restart the web server.

Setting up PortWeb

After copying the PortWeb plug-in to the appropriate folder, you’re ready to start setting up PortWeb via the PortWeb Administration Module. The Administration Module page allows you to tell PortWeb which Portfolio catalogs you want to make available on the Web. (A few “sample” served catalogs are filled in on the Administration Module page by default to show you how the path to a catalog should be specified. You can replace the sample data with your own, as outlined below.)

- 1 With your web server running, launch a web browser and enter the appropriate URL for the platform hosting the web server:

Windows

```
http://yourserver/scripts/PortWeb.dll  
?adminshow
```

(Where “yourserver” is the IP address or domain name of your web server.)

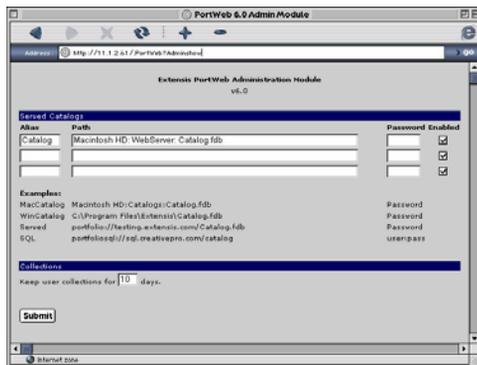
Macintosh

`http://yourserver/.PortWeb?adminshow`
(Where “yourserver” is the IP address or domain name of your web server.)

With WebStar, you can also access PortWeb’s Administration module via the Plug-ins page of the WebStar administration pages.

The web server software may prompt you for an administrator name and password. For more information on authenticating administrators, refer to the “Web Server Authentication” section on page 4.

2 The resulting web page should look something like:



3 In the alias field, type in a simple name for the catalog you want to serve. The alias should be a short, easily-remembered name by which a catalog will be referred to when specifying which catalog to search via PortWeb. Avoid using spaces and non-standard non-ANSI

characters (such as trademark symbols, ellipses, etc.) in this name, or you will have to take special steps to encode the alias in the your HTML pages.

Using aliases allows you to pass in a shorter request (i.e. “dogs” instead of a full path such as “D:\Portfolio\Files\Catalogs\dog database.fdb”) when specifying a catalog in a PortWeb query. Aliases also help you to avoid having to rewrite your HTML pages and templates every time the location of a catalog changes, because only the catalog’s alias (and not the whole path) is stored in the templates.

Served Catalogs	
Alias	Path
Catalog	Macintosh HD: WebServer: Catalog.fdb

Assign an alias by typing in a short name for your catalog

Tell PortWeb where to find your catalog by typing in the complete file path

4 In the Path field, type in the full path to the Portfolio catalog that you want to serve on the Web using Portweb. The catalog can be either a file on disk or a catalog being served by a Portfolio Server. Use the following rules to accurately specify the path to your catalog:

Local (on disk):

To reference a local path, simply type in the entire path name, such as:

Windows

`F:\Portfolio\Catalogs\dog.fdb`

or

Macintosh

`Macintosh HD:Portfolio:dog.fdb`

The catalog can be in any location on any disk accessible via the file system of the computer hosting PortWeb. Note that some Macintosh characters do not map to the ISO-standard character set used on the Internet. If your catalog's path has odd characters (such as copyright symbols, curly-quotes, etc.) you will probably need to change the path on disk to work over the Internet properly.

Served (using Portfolio Server):

To reference a Macintosh or Windows Portfolio catalog being served by a Portfolio Server, use the following format:

```
portfolio://128.0.0.01/dog.fdb
```

The word "portfolio:" must be all lower-case.

Served (using SQL Connect with Portfolio Server):

To reference a catalog being served on an SQL server, use the following format:

```
portfoliosql://192.168.0.1/dog.fdb
```

The word "portfoliosql:" must be all lower-case. Depending on the type of catalog being served, the filename may not have a suffix of *.fdb*

5 If your Portfolio catalog requires a password for *Reader*-mode access, type the password into the Password field. If there is no password for the Reader level of the catalog, simply leave the field blank. Note that because higher-level passwords work for lower levels (i.e. you can log into Reader mode with a Publisher password), it will work to use any of the four access levels' passwords in this field.

If your catalog is set up with user-based access, the proper password syntax to use when enabling catalogs is user:password. Example: If a user's

name is "Jenny" and the password is "duck" you would type "Jenny:duck" into the password field.



Type in the password only if it is required to access your catalog in Reader mode

Turn on the Enabled checkbox to make the specified catalog available via Portweb

6 Turn on the Enabled checkbox. This checkbox indicates whether the mapping is currently active. When this checkbox is unchecked, PortWeb releases the catalog. This is useful when you want to deactivate a catalog temporarily without having to delete and re-enter all the information on the entire line.

When a mapping is enabled, PortWeb opens a connection to the catalog. As a result, you cannot open a Portfolio catalog in Administrator mode or with the Portfolio Server while PortWeb is connected to that catalog.

Be aware that if you have a served catalog enabled in your list, the catalog will become inaccessible via PortWeb if you close that catalog on the Portfolio Server and then reopen that catalog on the Portfolio Server. As a result, you will get an empty page returned if you make a requests using this served catalog. To remedy this, return to the PortWeb Administration Module page and resubmit the page (with that mapping enabled).

If the Administration page displays an unchecked "Enabled" box after you submit it, verify that the path was entered correctly. Also verify that catalog is not opened exclusively by

a Portfolio client. If you are trying to access a served catalog, make sure you are using the appropriate syntax.

The administration module will display all the current mappings (both active and inactive) and two additional lines to add new mappings. If you want to enter more than two mappings in one session, simply enter the first two and click Submit to get a new form that contains the new mappings and two new lines in the form. To permanently delete a mapping, clear the Alias, Path, and Password entry fields and then click the Submit button.

7 Click the “Submit” button at the bottom of the Administration form. This instructs PortWeb to begin accepting requests for information from this catalog. If the Administration page is returned with the enable box disabled, check the path and verify that the catalog is not opened exclusively by a Portfolio client.

8 Test your PortWeb connection. You should now be able to access your Portfolio catalog via the Web. The easiest way to test this is to type a URL into your web browser that makes a direct query to Portweb.

In your web browser, type in the following URL (where “yourserver” is the IP address or domain name of your web server and “alias” is the alias name that you assigned to your catalog in the Administration Module.)

Macintosh:

```
http://yourserver/.PortWeb?quickfind&
catalog=alias
```

Windows

```
http://yourserver/scripts/PortWeb.dll
?quickfind&catalog=alias
```

If Portweb has been set up correctly, you should see a web page containing a four by four grid of thumbnails from your catalog with the filename displayed beneath each thumbnail.



This is the most basic form of PortWeb query. The documentation that follows will explain how you can:

- Create search pages that allow you to query Portweb for specific items within a catalog
- Create “response templates” that allow you to control the format of the HTML pages that result from a search
- Create “collections” of found results so that users can build “shopping carts” of found items when they search a catalog using Portweb.

Creating Search Pages

PortWeb does not generate a search form. Instead, you have total freedom to create search requests in any format you can conceive of in HTML. This allows PortWeb to integrate seamlessly with your existing web site. The search form you create can be as simple or as complex as your needs require. Two different commands provide for both simple and complex search queries.

QuickFind Command

The quickfind command is a simplified search mechanism for basic keyword searches. It performs a “Keywords starts with” search with the specified text. The acceptable parameters are shown in the table below:

QuickFind Command

Parameter	Description
quickfind	This is the name of the command, but also the parameter for passing in the keyword to search for.
catalog	Specifies the alias of the catalog to be used. The alias is mapped via the settings file to either a catalog on disk or a catalog being served by a Portfolio server.
template [OPTIONAL]	Specifies the name of the Export HTML template to be used for the response page. If this parameter is not passed in, a default Export HTML template will be used (a basic thumbnail view layout in a 4 x 4 grid).
sorton [OPTIONAL]	Identifies the field on which to sort the found set is sorted. The sort field must be an indexed single-value field. If it is not specified, the order of the record set is the order returned by the Find.
ascending [OPTIONAL]	Specifies the order in which the record set is sorted. A value of 1 indicates ascending, a value of 0 indicates descending. If the sorton parameter is not provided, this parameter is ignored. If the sorton parameter is provided and this parameter is not, the default sort order is 1 (ascending).
offset [OPTIONAL]	A positive integer indicating how many records have already been displayed from the collection. If the parameter is not specified, the display starts from the first record in the collection.

QuickFind Examples

This simple search looks for the records containing a keyword that starts with "Com" in the catalog aliased to "Sample" in the PortWeb settings file. Records containing the keywords "computer" and "communications" would both be returned.

```
<A HREF="/scripts/PortWeb.dll?quickfind=com&catalog=Sample">...</A>
```

This is essentially the same search, but using a form that allows the user to enter the keywords for which to search. This also shows all of the optional parameters being used.

```
<FORM ACTION="/scripts/PortWeb.dll?QuickFind" METHOD=POST >
  <INPUT NAME="quickfind" TYPE="Text" VALUE="">
  <INPUT NAME="template" TYPE="HIDDEN" VALUE="Simple">
  <INPUT NAME="sorton" TYPE="HIDDEN" VALUE="Filename">
  <INPUT NAME="ascending" TYPE="HIDDEN" VALUE="1">
  <INPUT NAME="offset" TYPE="HIDDEN" VALUE="10">
  <INPUT NAME="catalog" TYPE="HIDDEN" VALUE="Sample">
  <INPUT TYPE="submit">
</FORM>
```

Query Command

The query request is the core of PortWeb. This command causes a search to be performed in a particular Portfolio catalog, and the results to be returned in a particular layout. The values that can be passed in shown in the table below.

Query Command

Parameter	Description
query	This is the set of parameters that define the search.
catalog	Specifies the alias of the catalog to be used. The alias is mapped via the settings file to either a catalog on disk or a catalog being served by a Portfolio server.
template [OPTIONAL]	Specifies the name of the Export HTML template to be used for the response page. If this parameter is not passed in, a default Export HTML template is used (a basic thumbnail view layout in a 4 x 4 grid).
sorton [OPTIONAL]	Identifies an indexed, single-valued field on which the found set should be sorted. If it is not specified, the order of the record set is the order returned by the find.
ascending [OPTIONAL]	Specifies the order in which the record set is sorted. A value of 1 indicates ascending, a value of 0 indicates descending. If the sorton parameter is not provided, this parameter is ignored. If the sorton parameter is provided and this parameter is not, the default sort order is 1 (ascending).
offset [OPTIONAL]	A positive integer indicating how many records have already been displayed from the collection. If the parameter is not specified, the display starts from the first record in the collection.

Query Parameter

This parameter is a set of variables which define the search that should be executed in Portfolio.

Each clause of a query is defined by three variables: Field, Operator, and Value. Each clause (after the first) must be joined to the previous using a join variable (either “and” or “or”).

The acceptable values for each variable are shown in the table below.

Query Parameter

Variable	Acceptable Values
field	Any indexed field in the specified database. If an unindexed field is specified, the query will fail.
op	The following operators are allowed*: <ul style="list-style-type: none">• Matches / Does Not Match – Works with all fields except Date fields• Equals / Does Not Equal – Works only with Date fields• Greater Than / Greater Than or Equal To – Works only with Number fields• Less Than / Less Than or Equal To – Works only with Number fields• Starts With / Does Not Start With – Works only with String fields
value	Only works with Number fields.
join [OPTIONAL]	Used only when the query has more than one clause, this can have a value of either “and” or “or”. If this parameter is not specified, “and” will be used. You can join up to 10 clauses to create a multi-criteria search.

You can use all of the operators supported by the Portfolio client in PortWeb, however not all operators work with each field. If your operator does not work with a selected field, the search clause is skipped. If you're unsure whether an operator is supported with a specific field, you can verify it using the Portfolio client's find feature. Choose your fieldname in the Portfolio client's find dialog, then use the operator pull-down to see which operators are supported for that fieldname.

A query is validated by the plug-in based on the following criteria:

- All three variables — a field, an operator and a value — must exist for each clause.
- For every clause after the first one, a join variable must also be supplied (with the value being either “and” or “or”).
- Each field variable and operator variable must meet the rules set above.

If any of these validations fails for a particular clause, the clause is skipped. If validation fails for all clauses in a query, the plug-in returns the “No matching records found” page. If the validation succeeds, the remaining parameters are evaluated.

A query can contain a maximum of ten clauses.

Example:

The example below shows a two clause query command. (The comments are just for clarity). The various parameters can be hidden or exposed, depending on the site’s needs. In this example, the first clause’s value field is entered by the user, but all other values are hidden.

```
<FORM ACTION="/scripts/PortWeb.dll?query" METHOD=POST >
  <INPUT NAME="template" TYPE="HIDDEN" VALUE="Simple">
  <INPUT NAME="catalog" TYPE="HIDDEN" VALUE="Sample">

  <!-- This is the first clause, and prompts the user for a keyword -->
  <INPUT NAME="field" TYPE="HIDDEN" VALUE="Keywords">
  <INPUT NAME="op" TYPE="HIDDEN" VALUE="matches">
  <INPUT NAME="value" TYPE="TEXT" VALUE="">

  <!-- This is a second clause (with the required join) that is entirely hidden. -->
  <INPUT NAME="join" TYPE="HIDDEN" VALUE="and">
  <INPUT NAME="field" TYPE="HIDDEN" VALUE="Extension Win">
  <INPUT NAME="op" TYPE="HIDDEN" VALUE="matches">
  <INPUT NAME="value" TYPE="HIDDEN" VALUE="jpg">
  <INPUT TYPE="Submit">
</FORM>
```

GET and POST methods

In general, there are two mechanisms for communicating with a web server plug-in. These are known as the GET and POST methods.

Note that the GET method can be used in a form (simply by setting the Method to "GET"), but it is common practice to use the POST method in these situations.

The primary difference between the two methods is that POST allows for user intervention via a form for changing parameters, while the GET method has hard-coded parameters. Each of these methods is useful, depending on the situation. For example, if you wanted to have a simple button that automatically searches a particular catalog for a particular keyword, then the GET method is the simpler and more concise approach. However, if

you'd like to ask the user for particular parameters (such as the keywords to search for) then a form using the POST method is more applicable. In addition, the POST method can have unlimited request data and parameters, but the GET method can be limited to 4K of request information (depending on the browser).

Regardless of which method is used to communicate with PortWeb, the general structure of the request is the same. Each time a request is sent to the plug-in, the web server must first identify the plug-in, then the plug-in identifies the command to be executed and the parameters necessary for that command to work, as shown in the examples at the bottom of this page.

Plug-In Communication Methods

Method	Command
GET	Typically, the GET method is used as an embedded link, such as: <pre>...</pre>
POST	The POST method, on the other hand, is usually used inside an HTML Form, such as: <pre><FORM ACTION="/scripts/PortWeb.dll?QuickFind" METHOD=POST > <INPUT NAME="QuickFind" TYPE="TEXT" VALUE="cat"> <INPUT NAME="catalog" TYPE="HIDDEN" VALUE="WebTest"> <INPUT TYPE="submit" VALUE="Search"> </FORM></pre>

Identifying the Plug-In

The plug-in's location must always be provided in a request. This is what tells the web server to pass the request on to PortWeb. Plug-ins are identified differently on different platforms. Refer to the section corresponding to the platform of your web server. Throughout this section, the two platforms will be used interchangeably. Be sure to use the syntax appropriate for your platform, as shown in the tables.

Windows

With both the GET and POST methods, you must provide not only the name of the plug-in, but also the path to the plug-in on the server. The simplest mechanism for referencing the plug-in is to use a "root-relative" path, such as "/scripts/PortWeb.dll..." This indicates that the plug-in is located in the "scripts" directory at the root level of the web server.

Identifying the Plug-In—Windows

Method	Command
GET	Using the GET method, the plug-in is identified as the first part of the HREF: <pre>...</pre>
POST	Using the POST method, the plug-in is identified in the first half of the ACTION attribute: <pre><FORM ACTION="/scripts/PortWeb.dll?QuickFind" METHOD=POST ></pre>

Alternately, you could use a full URL to locate the plug-in, such as the one shown below. This is often used when the plug-in is running on a web server other than the one serving the web pages.

```
<A HREF="http://www.myserver.com/scripts/PortWeb.dll...></A>
```

Macintosh

Normally, the path to the plug-in is not required. This is because the plug-in is registered with the server when the web server launches.

Identifying the Plug-In—Macintosh

Method	Command
GET	Using the GET method, the plug-in is identified as the first part of the HREF: <hr/> <pre>...</pre> <hr/>
POST	Using the POST method, the plug-in is identified in the first half of the ACTION attribute: <hr/> <pre><FORM ACTION=".PortWeb?QuickFind" METHOD=POST ></pre> <hr/>

However, if the plug-in is on a separate web server other than the one serving the web pages, you would use the full URL. For example:

```
<FORM ACTION="http://otherserver.com/.PortWeb?QuickFind" METHOD=POST >
```

PortWeb Commands

The commands currently supported in PortWeb are shown in the table below.

The command is always passed in immediately following the plug-in location. The question mark (?) is used to indicate the beginning of the request variables. The first variable indicates which request (**quickfind**, **query**, etc.) to perform.

Using the GET method, the command immediately follows the plug-in location.

```
<A HREF="/scripts/PortWeb.dll?quickfind=cat&catalog=Sample">...</A>
```

Using the POST method, the command is identified in the second half of the ACTION attribute:

```
<FORM ACTION="/scripts/PortWeb.dll?quickfind" METHOD=POST >
```

PortWeb Commands

Command	Description
quickfind	A very simple search command for doing basic keyword, filename and description searches.
query	A more flexible search command for creating complex searches.
add	Adds one or more records to a user's collection (shopping cart).
show	Displays the user's collection.
remove	Removes one or more records from a user's collection.
adminshow	Displays the Administration module for PortWeb.

PortWeb Parameters

Each command requires one or more parameters. These are all the variables that affect what the command does. For example, two of the most important parameters for any request are the name of the catalog to search and the response template to be used.

Using the GET method, parameters are specified directly within the HREF, and are entered as value pairs (a parameter and its value). Each value pair is separated by an ampersand (&). The parameter is separated from the its value by an equals sign (=). In the following example, the parameter "catalog" is passed in with the value "Sample."

```
<A HREF="/scripts/PortWeb.dll?quickfind=cat&catalog=Sample">...</A>
```

Using the POST method, the parameters are defined using form INPUT tags. In the following example, the parameter "catalog" is passed in with the value "Sample." In this case, the parameter is hidden, so that it cannot be modified by the user. Other INPUT types (such as "TEXT") could be used to get information from the user. For more information on setting up HTML forms, please refer to an HTML reference guide.

```
<FORM ACTION="/scripts/PortWeb.dll?quickfind" METHOD=POST >  
  <INPUT NAME="catalog" TYPE="HIDDEN" VALUE="Sample">  
</FORM>
```

Creating Response Templates

The response to a PortWeb command is formatted by a response template stored in the Portfolio catalog. The response templates are the same templates that are used by the Portfolio client in the Export HTML dialog. Response templates are created in the Portfolio 6.0 client by creating new templates using the Create Web Pages command. See pages 106-108 and pages 111-112 in the *Portfolio 6 User Guide*.

The Default Template

The simplest response template is simply no template at all. If no response template is specified in a PortWeb command, a default response is generated. For example, the following request will use the default template because no template parameter is specified.

Macintosh

```
http://yourserver/.PortWeb?quickfind=com&catalog=Sample
```

Windows

```
http://yourserver/scripts/PortWeb.dll?quickfind=com&catalog=Sample
```

(Where “yourserver” is the IP address or domain name of your web server.)

The response is a four by four grid of thumbnails with the filename displayed beneath each thumbnail. For some purposes, this default response may be sufficient.

Creating a Simple Template

In most situations, a customized response template is necessary. To create simple templates, you can customize your view in Portfolio and then add a new template to the collection of HTML Templates in the Create Web Pages dialog box. The template HTML displayed in the Create Web Pages dialog box is an HTML approximation of the gallery you have created.

Example:

- 1 Open a catalog in the Portfolio 6.0 client.
- 2 Customize the font, spacing and background color used in the current gallery window.
- 3 Choose Item > Create Web Pages.
- 4 From the Create Web Pages dialog, click the Add... button.
- 5 The Edit HTML dialog is displayed. In this will be the code Portfolio generated that approximates the view in Portfolio.
- 6 Enter a name for the template, such as “blue.”
- 7 Enter grid settings, such as four rows by one column. In the case of Record Views, you typically would want one column.
- 8 Click Save.
- 9 Upon returning to the the Create Web Pages dialog, you'll see the template was added to the

list of templates in the catalog. Click Close to return to Portfolio.

To access this template via the web, it is simply a matter of supplying the template name as a parameter. So, building on the previous example, we can expand it to:

Macintosh

```
http://yourserver/.PortWeb?quickfind=com&catalog=Sample&template=blue
```

Windows

```
http://yourserver/scripts/PortWeb.dll?quickfind=com&catalog=Sample&template=blue
```

(Where “**yourserver**” is the IP address or domain name of your web server.)

Collections

Collections (also known as “shopping carts”) are a common feature of many web sites, and allow users to keep track of a particular set of items. This is particularly useful in the context of an image database, in which a user may want to gather a set of images from a number of different searches.

PortWeb has built-in support for collections. PortWeb creates a shopping cart for any user who wants to track items (assuming the page is designed to make use of shopping carts). PortWeb can maintain a user’s shopping cart across multiple sessions, and can maintain separate carts for each catalog a user visits.



Although collections are referred to as “shopping carts” Portweb does not include any components for e-commerce. An e-commerce solution must be purchased separately.

Collection record sets are stored in a proprietary database file named “PortWeb.wdb”. The Administrator has the ability to set how long collections are kept in the database. Below are the various commands. Note that the collection requests only work correctly for users who allow their browsers to accept cookies (see the end of this section).

Note: Collections are stored separately for different catalogs. In other words, items from one catalog cannot be added to the same shopping cart as items from a different catalog.

Working with Collections

The common user experience with a collection in PortWeb is:

1. The user submits a search via the query command.
2. The user is returned a web page generated by the query request.
3. On the page is a button reading “Add to my shopping cart.”
4. Clicking this generates a Collection request to add one or more records. If no collection currently exists for the current user in the current catalog (based on the cookie received in the query), a new collection record is created.
5. The result of the Add command is a newly generated page showing the current collection, and a cookie is placed on the user’s system.
6. From this shopping cart page, the user can choose to remove items from the cart.

To add collection functionality to a site powered by PortWeb, you can either use collection macros, or use the collection commands directly. Normally it is easier to use the macros, as PortWeb will generate much of the necessary HTML automatically. However, there may be times when the flexibility provided by using the collection commands is desired.

Collection Templates

Collection templates are not really any different than any other response template. The primary difference is that they are displaying results

from a user's collection set, instead of results from a particular query. Often the same response templates can be used for both a query and a collection. Of course, if you want to provide the users the ability to manipulate the collection, you'll need to create collection templates that make use of the Add and Remove commands.

Collections setting in Administration Module

The Collections section of the Portweb Administration Module (described on pages 5-8) controls the duration for which Collection records should be kept in the database. Simply specify the number of days for which the users' collections should be retained. Bear in mind that the collection database's file size will increase the longer you retain these records.

Collection Commands

Add Command

The **add** command adds one or more records to the user's collection set. If no collection set for the user of the particular catalog exists, then a new collection set is created and a cookie is returned to the user. The parameters specified are:

Add Command

Parameter	Description
catalog	Specifies the alias of the catalog is to be used. The alias is mapped via the settings file to either a catalog on disk or a catalog being served by a Portfolio server.
rid	Specifies the record ID in the catalog. Record IDs are unique within a catalog. This can be one record or a list of records (e.g., ...&rid=70&rid=75&rid=85&...). Any invalid RIDs are ignored.
template [OPTIONAL]	Specifies the alias of the template to be used. If this parameter is not passed in, a default Export HTML template is used (a basic thumbnail view layout).
sorton [OPTIONAL]	Identifies an indexed, single-valued field on which the found set should be sorted. If it is not specified, the order of the record set is the order returned by the Find.
ascending [OPTIONAL]	Specifies the order in which the record set is sorted. A value of 1 indicates ascending, a value of 0 indicates descending. If the sorton parameter is not provided, this parameter is ignored. If the sorton parameter is provided and this parameter is not, the default sort order is 1 (ascending).
offset [OPTIONAL]	A positive integer indicating how many records have already been displayed from the collection. If the parameter is not specified, the display should start from the first record in the collection.

Remove Command

The **remove** command removes one or more records from the user's collection set. If no cookie exists, or if the ID in the user's cookie does not match any records in the collections database, or if the IDs in the cookie do not match the catalog specified in the request, then an error is returned to the user on the "No matching records found" page. The parameters specified are:

Remove Command

Parameter	Description
catalog	Specifies the alias of the catalog is to be used. The alias is mapped via the settings file to either a catalog on disk or a catalog being served by a Portfolio server.
rid	Specifies the record ID in the catalog. Record IDs are unique within a catalog. This can be one record or a list of records (e.g., ...&rid=70&rid=75&rid=85&...). Any invalid RIDs are ignored.
template [OPTIONAL]	Specifies the alias of the template to be used. If this parameter is not passed in, a default Export HTML template is used (a basic thumbnail view layout).
sorton [OPTIONAL]	Identifies an indexed, single-valued field on which the found set should be sorted. If it is not specified, the order of the record set is the order returned by the Find.
ascending [OPTIONAL]	Specifies the order in which the record set is sorted. A value of 1 indicates ascending, a value of 0 indicates descending. If the sorton parameter is not provided, this parameter is ignored. If the sorton parameter is provided and this parameter is not, the default sort order is 1 (ascending).
offset [OPTIONAL]	A positive integer indicating how many records have already been displayed from the collection. If the parameter is not specified, the display should start from the first record in the collection.

Show Command

The `show` command displays the records in a user's collection set. The parameters are:

Show Command

Parameter	Description
<code>catalog</code>	Specifies the alias of the catalog is to be used. The alias is mapped via the settings file to either a catalog on disk or a catalog being served by a Portfolio server.
<code>rid</code>	Specifies the record ID in the catalog. Record IDs are unique within a catalog. This can be one record or a list of records (e.g., <code>...&rid=70&rid=75&rid=85&...</code>). Any invalid rids are ignored.
<code>template [OPTIONAL]</code>	Specifies the alias of the template to be used. If this parameter is not passed in, a default Export HTML template is used (a basic thumbnail view layout).
<code>sorton [OPTIONAL]</code>	Identifies an indexed, single-valued field on which the found set should be sorted. If it is not specified, the order of the record set is the order returned by the Find.
<code>ascending [OPTIONAL]</code>	Specifies the order in which the record set is sorted. A value of 1 indicates ascending, a value of 0 indicates descending. If the <code>sorton</code> parameter is not provided, this parameter is ignored. If the <code>sorton</code> parameter is provided and this parameter is not, the default sort order is 1 (ascending).
<code>offset [OPTIONAL]</code>	A positive integer indicating how many records have already been displayed from the collection. If the parameter is not specified, the display should start from the first record in the collection.

Cookies

Whenever a collection request (**add**, **show**, **remove**) is successful, a cookie containing the collection's id number is sent from the web server to the client's browser. If the cookie is accepted, it is stored on the user's machine, and sent back to PortWeb for every subsequent request.

Users can refuse individual cookies, or set their browsers to refuse all cookies. If cookies are refused, every add request will create a new, one-item collection in PortWeb's collection database. In this case, the results of all collection requests (**add**, **show**, and **remove**) will act as if the request had been applied to an empty cart.

Cookies set by PortWeb have an expiration date that matches the plug-in's collection expiration setting.

PortWeb Templates

PortWeb templates are HTML templates that are used to format and display the results that PortWeb produces in response to a request.. They can contain virtually any HTML code of which you can conceive.

PortWeb does not analyze the HTML you enter, other than to verify that you have used legitimate macro names. Therefore, any HTML you can create can be used, including JavaScript, DHTML, XML, etc.

There are three key aspects to an Export HTML template that are unique to PortWeb: Template Blocks, Template Macros, and Macro Blocks.

Template Blocks

PortWeb's HTML templates are made up of three blocks. You can define HTML for the Header, Item, and Footer. The Header section is placed at the beginning of each generated page, the item section is inserted into each table cell, and the footer is placed at the end of each generated page. In Portfolio's Edit Template dialog (accessed by choosing Item > Create Web Pages), you can either type the HTML directly into each block or paste it in using Paste from the Edit menu.

The following rules apply to each block in a PortWeb template:

Header:

1. The header must contain the <HTML> tag and the <BODY> tag.
2. Because the grid layout is generated as a table, the Header must end with a <TABLE> tag.

Footer:

1. The Footer must start by closing out the table with a </TABLE> tag.
2. The Footer should close out with </BODY> and </HTML> tags.

Item:

1. The item block should contain all of the formatting for a particular table cell, including the opening <TD> tag and the closing </TD> tag.
2. PortWeb will generate the row tags (<TR> and </TR>) on its own, based upon the grid dimensions. Do not include these tags in any of the cells (unless you are adding your own rows at the top or bottom of the table).

Template Macros

Portfolio uses a macro language in the HTML Templates to allow substitution of variable data at the time the HTML files are actually generated. All macros start and end with a % sign, so the proper use would be %macro%. To display a "%" sign in the actual HTML, use two consecutive % signs in the template (i.e. "%%" in the template produces one "%" in the generated output). Macros are case-insensitive.

The complete list of acceptable macros is shown in the table on the next page:

PortWeb HTML Template Macros

Macro Name	Description
<code>%fieldname%</code>	Any field available to the user can be inserted into the item block. At the time of export, the value of the field for the particular record being processed is substituted for the macro.
<code>%thumbnail%</code>	Substitutes a thumbnail request. Note that this is different functionality than the existing Export HTML mechanism (which was putting a path to disk).
<code>%RID%</code>	Inserts the Record ID for the particular record. The Record ID is unique in the catalog and is used by commands such as Thumbnail, Add, and Remove to reference specific records in the catalog.
<code>%prev%</code>	Substitutes a relative link to the previous HTML file generated in the series. If the file being generated is the first file, an empty string is substituted.
<code>%next%</code>	Substitutes a relative link to the next HTML file to be generated in the series. If the file being generated is the last file, an empty string is substituted.
<code>%pagelist%</code>	Generates HTML for a set of page number links (i.e. 1 2 3 4 5)
<code>%page%</code>	Inserts the number of the current page into the HTML.
<code>%totalpages%</code>	Inserts the count of the total number of pages that could be generated, given the grid size and the count of the found record set.
<code>%totalitems%</code>	Inserts the count of the total number of found records for the submitted query.
<code>%collectionadd%</code>	Generates the HTML for a Collection Add request.
<code>%collectionremove%</code>	Generates the HTML for a Collection Remove request.
<code>%Ccollectionshow%</code>	Generates the HTML for a Collection Show request.

PortWeb HTML Template Macros

Macro Name	Description
<code>%original%</code>	<p>Substitutes the full path to the original file cataloged in Portfolio. Embedding this macro in an HREF creates a link to the original file. The web server must have access to the original file for this macro to function properly.</p> <p>EXAMPLE: <code> View Original </code></p>
<code>%preview%</code>	<p>Substitutes the path to the disk preview JPEG version of a cataloged item, <i>if a disk preview exists</i>. This macro can be used to display a low-resolution version of an image on a web page, or to display a JPEG preview of an item that is in a format that cannot be displayed in a web browser.</p>

Special Macros

A few sets of macros behave differently, depending on whether the Portfolio client or PortWeb is generating the results.

Page Macros

[%prev%, %next%, %page% and %pagelist%]

The HTML generated by the Portfolio client for %Prev%, %Next%, and %PageList% macros generate links to other HTML pages on disk. When these macros are handled by PortWeb, the generated HTML is instead new quickfind or query commands with the appropriate offset parameter.

For example, a template that generates two rows by four columns will use the following macro:

```
<A HREF = "%Next%">
```

to generate this HTML:

```
<A HREF=".PortWeb?quickfind=
test&catalog=Sample&template=Simple
&offset=8">
```

The catalog name, quickfind value, template name, and offset are all automatically determined based on the original request. When the user clicks on the link, a new quickfind command is submitted to PortWeb that instructs PortWeb to start generating results with the ninth item. This powerful feature means PortWeb only generates results for one page at a time, meaning response times are kept to a minimum.

While you could conceivably create these links manually, Portfolio automates all of this for you.

Thumbnail Macro

[%thumbnail%]

The %thumbnail% macro, when handled by PortWeb, generates a Thumbnail command and fills in the appropriate parameters.

For example, a template that contains the following macro:

```
<IMG SRC="%thumbnail%">
```

will generate the following HTML:

```
<IMG SRC="PortWeb.dll?
thumbnail=20&catalog=WebTest">
```

The Record ID specified for the catalog and the catalog name are automatically determined based on the original request.

Thumbnail Command

The thumbnail request is used to request a JPEG data stream of a particular thumbnail. This allows PortWeb to stream thumbnail data from a particular record in a catalog back to the web browser without ever having to write the data to disk first. Typically, these requests will be embedded in the HTML results from the query request. This will cause the browser to submit these requests back to PortWeb, allowing it to return each thumbnail as an individual JPEG stream.

The thumbnail command should always be part of an image link:

```
<IMG SRC="PortWeb.dll?
thumbnail=80&catalog=WebTest">
```

The two parameters that must be supplied are shown below.

Usage

There are two Export HTML macros you can use to supply the record id: %thumbnail% and %rid%. The %thumbnail% macro provides the entire path for the image's SRC parameter, while the %RID% supplies only the record's id.

Examples:

```
<IMG SRC="%thumbnail%">
```

```
<IMG SRC="PortWeb.dll?thumbnail=
%rid%&catalog=Sample">
```

Normally it is easier to use the %thumbnail% macro, since it automatically encodes the RID and catalog name, but there may be times when you want to use the additional flexibility of manually establishing the other parameters.

Thumbnail Command

Parameter	Description
thumbnail	Specifies the record's unique ID in the catalog. This parameter must appear first when doing a GET request.
catalog	Specifies the alias of the catalog is to be used. The alias is mapped via the settings file to either a catalog on disk or a catalog being served by a Portfolio server.

Collection Macros

```
[%CollectionAdd%, %CollectionRemove%,  
and %CollectionShow%]
```

These commands have no purpose when using Export HTML with the Portfolio client, as they are dynamic commands that are built so that PortWeb can remember users' selections.

Macro Blocks

Export HTML allows for markers to delineate the beginning and end of "macro blocks." Macro blocks define the beginning and end of a section of HTML that surrounds a Portfolio macro. If the macro evaluates to an empty value, then none of the HTML text within the macro block is generated. Macro blocks are defined using HTML comment tags in the following structure:

```
<!--Port4-%macro%-->
```

and

```
<!--/Port4-%macro%-->
```

where the first comment opens the block and the second comment closes the block. As PortWeb processes the template, each time it encounters an open comment, it evaluates the macro defined within. If the macro has a value of nil (for example, if the field is empty), PortWeb skips the HTML code until the corresponding close comment is found (or the end of the section is reached).

The most common use for the macro blocks is the "Previous" and "Next" macros, so that additional information does not get generated. Take the following example:

```
<!--Port4-%prev%-->  
<A HREF="%prev%">Previous Page</A>  
<!--/Port4-%prev%-->
```

When PortWeb processes this section for page one of a search, if it were on the first page, the %prev% macro would evaluate to "nothing." Therefore, PortWeb would not generate any text until the close comment is encountered.

Another example would be to add a field name in front of a field macro, but not display the field name if there was no value for the field:

```
<!--Port4-%filename%-->  
File Name: %filename% <BR>  
<!--/Port4-%filename%-->
```

A more elaborate possibility would use the macro of one field as a logic flag for determining whether to display a completely different piece of information:

```
<!--Port4-%HasSource%-->  
<A HREF="%path%">%filename%</A>  
<!--/Port4-%HasSource%-->
```

The preceding example would insert a hyperlink to the source file, but only if there was a value in the field "HasSource."

Note that it is possible to nest macro blocks, but it may result in unanticipated results if the blocks are not properly nested or if the blocks are not properly terminated with a close comment, since PortWeb simply stops generating text until it encounters the appropriate close block or the end of the section.

Special Macro Blocks

Several macro blocks are set to evaluate to empty in particular situations:

- %prev% will evaluate to empty if the page being generated is the first page in a record set. This is useful for omitting a "Previous" link

from the first page of a set.

- `%next%` will evaluate to empty if the page being generated is the last page in a record set. This is useful for omitting a “Next” link from the last page of a set.
- `%collectionadd%`, `%collectionremove%`, and `%collectionshow%` will evaluate to empty if the page is being generated by the Portfolio 6.0 client. Since these commands only work with PortWeb, this can allow you to use the same templates with both the Portfolio client and PortWeb.
- `%totalitems%` will evaluate to empty if no items are returned by a request. This is useful for generating HTML when there is a found set and you want to hide text, formatting, or buttons when no records are returned by a search.

Working with Templates on Disk

The Portfolio 6.0 client has the capability to save Portfolio’s HTML Templates to disk. This is particularly useful if you are working with larger amounts of HTML code, as you can then edit the HTML in an HTML-savvy text editor. In addition, it makes it much easier to move templates between catalogs or share templates with other users. Note that Portfolio’s custom formatting makes it difficult to work with these templates in visual page editors, such as Macromedia Dreamweaver or Adobe PageMill.

Template File Structure

When templates are written to files on disk, Portfolio inserts specific comment tags to define the beginning and end of the three template blocks. The tags are:

```
Header Start <!--Port4-Header-->
          End <!--/Port4-Header-->
Item    Start <!--Port4-Item-->
          End <!--/Port4-Item-->
Footer Start <!--Port4-Footer-->
          End <!--/Port4-Footer-->
```

In between each set of tags is the contents of each block. Other than these block tags, the HTML is identical to the HTML displayed in the Edit HTML dialog.

When the template is imported, Portfolio looks for the same tag pairs to determine into which blocks to place the HTML code. The tags themselves are stripped.

Exporting Templates

To export a template, simply click the Export... button in the Edit HTML dialog. You will be prompted to name the file. Once it is exported, you can open the file in any text editor.

Importing Templates

To import a template, simply click the Import... button in the Edit HTML dialog. You will be prompted to locate a file to import. Once you select a file, Portfolio scans it for the block tags and imports the information into the appropriate fields. If the appropriate start and end tags are not found, Portfolio will not import any of the data. All three sections do not need to be present in the file; if a section is missing, the block in the Edit HTML dialog will be untouched.

Note: Portfolio uses the "%" sign to denote special macros, and will not handle "%" symbols that may be part of your HTML (such as "width=100%"). When importing HTML generated by other

applications, convert each "%" sign in the HTML source file to two consecutive "%" signs in the template (i.e. "width=100%" in the source file should be changed to "width=100%%" in the template).

Using Collection Macros

The three collection macros are %collectionadd%, %collectionremove%, and %collectionshow%. These macros behave differently depending on whether they are placed in the Item block or in the Header or Footer blocks.

Collection Macros in the Item Block

When the macros are placed in the Item block, a partial GET request is generated. For example, entering the %collectionadd% macro results in the following HTML being generated:

```
.PortWeb?add&rid=75&catalog=Sample
```

For this to be useful, the macro needs to be inserted into an anchor of some type. For example, consider the following template code:

```
<A HREF = "%collectionadd%  
&template=myCart">Add to Cart</A>
```

The resulting generated HTML is:

```
<A HREF = ".PortWeb?add&rid=75  
&catalog=Sample&template=myCart">  
Add to Cart</A>
```

This is then a valid link that, when clicked, would add Record ID 75 to the user's collection and display the contents of the cart using the "myCart template." Note that the "&template=myCart" parameter had to be specified separately, as the system does not

know what template you want to use to display the cart. If the template parameter had been omitted, the default 4x4 thumbnail response would have been used. For more details on the parameters that are used by each collection command, refer to the section below, titled "Collection Commands."

The %collectionremove% and %collectionshow% macros behave in much the same way when placed in the Item block. The %Collectionremove% macro will automatically generate the rid and catalog parameters; you must supply the template and other optional parameters. The %collectionshow% macro will automatically generate the catalog parameter; again, you must supply the template and other optional parameters.

Collection Macros in the Header and Footer Blocks

When the collection macros are placed in the Header or Footer blocks, the generated HTML is different. This is because these commands have different meaning in this context, because there are no particular records that can be referenced from these locations.

Instead, PortWeb generates the start of an HTML Form block. For example, placing the %collectionadd% macro in the header results in the following HTML being generated:

```
<FORM ACTION=".PortWeb?add"  
METHOD=POST>  
<INPUT TYPE=HIDDEN NAME=catalog  
VALUE="Sample">
```

While at first glance this may not appear particularly useful, this is actually a very valuable feature. Because PortWeb is capable of

handling multiple items in an Add or Remove command, you can set up the template so that multiple records can be submitted in one form. For example, add the following line to the item block.

```
<INPUT TYPE="checkbox" NAME="rid"
VALUE=%rid%>Add to Cart
```

And then place a Submit button and a close Form tag in the footer.

```
<INPUT TYPE="submit" NAME="send"
VALUE="Add Selected Items">
<INPUT TYPE=HIDDEN NAME=template
VALUE="Sample"> </FORM>
```

The generated HTML will have a checkbox for each item. If you check a set of these boxes and click the "Add Selected Items" button, all of the checked records will be added to the shopping cart and the results will be displayed using the "Sample" template.

PortWeb Error Messages

Below is a guide to the error messages PortWeb may generate and their possible causes:

Error:

"There are no records matching your query"

Possible causes:

- Invalid search criteria. You may have performed a search for a keyword or field that does not exist in the catalog
- There are no matching records containing the specified keyword.

Error:

"PortWeb could not understand your request"

Possible causes:

- Incorrect syntax used in a quickfind command or query. Check syntax of PortWeb requests to make sure they conform to the syntax outlined earlier in this document.

Error:

"PortWeb could not find the specified catalog"

Possible causes:

- There is no catalog alias specified in the query.
- A mis-spelled or incorrect catalog alias is specified in the query.
- A catalog and alias are entered in the PortWeb Administration Module, but not enabled. Turn on the Enabled checkbox to make the catalog accessible.