

# Portfolio Page Skeleton

see portMacros.m4 for more details

PORTFOLIO_HEADING \$1	
PORTFOLIO_HEADING \$2	
PORTFOLIO_NAV	
LINK_IMAGE( PORTFOLIO_RECORD \$2, PORTFOLIO_RECORD \$3 )	PORTFOLIO_RECORD \$1
	PORTFOLIO_RECORD \$4
	PORTFOLIO_RECORD \$5
LINK_IMAGE( PORTFOLIO_RECORD \$2, PORTFOLIO_RECORD \$3 )	PORTFOLIO_RECORD \$1
	PORTFOLIO_RECORD \$4
	PORTFOLIO_RECORD \$5
PORTFOLIO_FOOTNOTE	

```
1: Path to files:
2:
3: ~/www - top level of web space, accessed via http://drdan.dmazzola.com:8080/~you/
4:
5: ~/www/port - phase I of final project
6: ~/www/port2 - phase II of final project
7:
8: Using ~/www/port2 as you 2nd phase of the final project, the layout
9: of files would follow this pattern:
10:
11: ~/www/port2/
12:
13: ./index.html           - portfolio navigation page generated by genGal
14: ./index.m4             - intermediate m4 file generated by awkPort
15: ./port.conf            - sample port.conf for this Portfolio of photos
16:
17: ./lib
18: ./lib/port.conf.skel    - skeleton of proposed port.conf
19: ./lib/portSkel.html     - web page showing skeleton of page layout
20: ./lib/filemap.txt       - this file
21:
22:
23: ./Daniel_Jacob
24: ./Daniel_Jacob/index.html - portfolio navigation page generated by genPort
25: ./Daniel_Jacob/index.m4   - intermediate m4 file generated by awkPort
26: ./Daniel_Jacob/port.conf  - Daniel_Jacob portfolio configuration file
27: ./Daniel_Jacob/styles     - SYMBOLIC Link to ../../styles
28:
29: ./Daniel_Jacob/DJ_month00
30: ./Daniel_Jacob/DJ_month00/images
31: ./Daniel_Jacob/DJ_month00/images/orig
32: ./Daniel_Jacob/DJ_month00/images/orig/IMG_1229.JPG
33: .....
34: ./Daniel_Jacob/DJ_month00/images/orig/IMG_1295.JPG
35: ./Daniel_Jacob/DJ_month00/images/small
36: ./Daniel_Jacob/DJ_month00/images/small/IMG_1229.JPG
37: .....
38: ./Daniel_Jacob/DJ_month00/images/small/IMG_1295.JPG
39: ./Daniel_Jacob/DJ_month00/images/thumb
40: ./Daniel_Jacob/DJ_month00/images/thumb/IMG_1229.JPG
41: .....
42: ./Daniel_Jacob/DJ_month00/images/thumb/IMG_1295.JPG
43: ./Daniel_Jacob/DJ_month00/m4
44: ./Daniel_Jacob/DJ_month00/m4/0000.m4
45: .....
46: ./Daniel_Jacob/DJ_month00/m4/0007.m4
47: ./Daniel_Jacob/DJ_month00/etc
48: ./Daniel_Jacob/DJ_month00/etc/captions.txt
49: ./Daniel_Jacob/DJ_month00/0000.html
50: .....
51: ./Daniel_Jacob/DJ_month00/0007.html
52: ./Daniel_Jacob/DJ_month00/index.html
53:
54: ./Daniel_Jacob/DJ_month02
55:
56: ./Daniel_Jacob/DJ_month02/images
57: ./Daniel_Jacob/DJ_month02/images/orig
58: ./Daniel_Jacob/DJ_month02/images/orig/IMG_1421.JPG
59: .....
60: ./Daniel_Jacob/DJ_month02/images/orig/IMG_1430.JPG
61: ./Daniel_Jacob/DJ_month02/images/small
62: ./Daniel_Jacob/DJ_month02/images/small/IMG_1421.JPG
63: .....
64: ./Daniel_Jacob/DJ_month02/images/small/IMG_1430.JPG
65: ./Daniel_Jacob/DJ_month02/images/thumb
66: ./Daniel_Jacob/DJ_month02/images/thumb/IMG_1421.JPG
67: .....
```

```
68: ./Daniel_Jacob/DJ_month02/images/thumb/IMG_1430.JPG
69: ./Daniel_Jacob/DJ_month02/m4
70: ./Daniel_Jacob/DJ_month02/m4/0000.m4
71: .....
72: ./Daniel_Jacob/DJ_month02/m4/0004.m4
73: ./Daniel_Jacob/DJ_month02/etc
74: ./Daniel_Jacob/DJ_month02/etc/captions.txt
75: ./Daniel_Jacob/DJ_month02/0000.html
76: .....
77: ./Daniel_Jacob/DJ_month02/0004.html
78: ./Daniel_Jacob/DJ_month02/index.html
79:
80: ./Vacations
81:
82: <These next three files do not exist in original distribution, you will
83: make these yourself, starting with port.conf and then running genGal
84: will make index.m4 and index.html>
85:
86: ./Vacations/index.html - portfolio navigation page generated by genPort
87: ./Vacations/index.m4   - intermediate m4 file generated by awkPort
88: ./Vacations/port.conf  - Vacations portfolio configuration file
89:
90: ./Vacations/Honeymoon
91: ./Vacations/Honeymoon/images
92: ./Vacations/Honeymoon/images/orig
93: ./Vacations/Honeymoon/images/orig/img_0075.jpg
94: .....
95: ./Vacations/Honeymoon/images/orig/img_0135.jpg
96: ./Vacations/Honeymoon/images/small
97: ./Vacations/Honeymoon/images/small/img_0075.jpg
98: .....
99: ./Vacations/Honeymoon/images/small/img_0135.jpg
100: ./Vacations/Honeymoon/images/thumb
101: ./Vacations/Honeymoon/images/thumb/img_0075.jpg
102: .....
103: ./Vacations/Honeymoon/images/thumb/img_0135.jpg
104: ./Vacations/Honeymoon/etc
105: ./Vacations/Honeymoon/m4
106: ./Vacations/Honeymoon/m4/0000.m4
107: .....
108: ./Vacations/Honeymoon/m4/0009.m4
109: ./Vacations/Honeymoon/0000.html
110: .....
111: ./Vacations/Honeymoon/0009.html
112: ./Vacations/Honeymoon/index.html
113: ./Vacations/Seattle
114: ./Vacations/Seattle/images
115: ./Vacations/Seattle/images/orig
116: ./Vacations/Seattle/images/orig/img_0726.jpg
117: .....
118: ./Vacations/Seattle/images/orig/img_0742.jpg
119: ./Vacations/Seattle/images/small
120: ./Vacations/Seattle/images/small/img_0726.jpg
121: .....
122: ./Vacations/Seattle/images/small/img_0742.jpg
123: ./Vacations/Seattle/images/thumb
124: ./Vacations/Seattle/images/thumb/img_0726.jpg
125: .....
126: ./Vacations/Seattle/images/thumb/img_0742.jpg
127: ./Vacations/Seattle/etc
128: ./Vacations/Seattle/m4
129: ./Vacations/Seattle/m4/0000.m4
130: .....
131: ./Vacations/Seattle/m4/0010.m4
132: ./Vacations/Seattle/0000.html
133: .....
134: ./Vacations/Seattle/0010.html
```

```

135: ./Vacations/Seattle/index.html
136:
137: ./bin
138: ./bin/awkPort          - called from genPort - parses port.conf
139: ./bin/genPort          - generates index.html portfolio files from port.conf
140: ./bin/genNC            - sample program to generate NavigationCell data
141:
142:
143: ./glyph
144: ./glyph/left_arrow.gif
145: ./glyph/right_arrow.gif
146:
147: ./macros
148: ./macros/basicHTMLMacros.m4
149: ./macros/genPagesMacros.m4
150: ./macros/portMacros.m4
151:
152: ./styles/
153: ./styles/project.css      - css for genPages html
154: ./styles/portStyles.css   - css for genPort html
155: ./styles/portSkelStyles.css - css for the portSkel.html file
156:
157:
158: Navigating a structured web site:
159: As an example, suppose that at the top level there are two portfolios
160:
161: The top or Home level has two portfolios:
162:     1) Pictures of Daniel Jacob
163:     2) Pictures of Vacations
164:
165: In turn, the portfolio of Daniel Jacob pictures has two sub portfolios
166:     1.1) Pictures of DJ @ 0 months old
167:     1.2) Pictures of DJ @ 2 months old
168:
169: And the Vacations portfolio has two sub portfolios:
170:     2.1) Pictures of your Honeymoon
171:     2.2) Pictures of your trip to Seattle
172:
173: So, starting at your top level, say it is called:
174:
175:     ~/www/port2
176:
177: There would be the following directory structure:
178:
179:     ~/www/port2/
180:         Danel_Jacob
181:         Vacations
182:
183: Expanding even further:
184:
185:     ~/www/port2/
186:         Daniel_Jacob/
187:             DJ_month_00/
188:             DJ_month_02/
189:         Vacations/
190:             Honeymoon/
191:             Seattle/
192:
193: Expanding even futher:
194:
195:     ~/www/port2/
196:         Daniel_Jacob/
197:             DJ_month_00/
198:                 0000.html
199:                 000N.html
200:                 etc/
201:                 images/

```

```

202:             m4/
203:                 index.html
204:         DJ_month_02/
205:             0000.html
206:             000N.html
207:             etc/
208:             images/
209:             m4/
210:                 index.html
211:
212:         Vacations/
213:             Honeymoon/
214:                 0000.html
215:                 000N.html
216:                 etc/
217:                 images/
218:                 m4/
219:                     index.html
220:
221:             Seattle/
222:                 0000.html
223:                 000N.html
224:                 etc/
225:                 images/
226:                 m4/
227:                     index.html
228:
229: Now, collapsing back one level, you need to have a port.conf
230: file that is the source of some program that will create
231: and index.html file that will let you navigate to various levels.
232: Put a port.conf file at each location specified below:
233:
234:     ~/www/port2/
235:         port.conf <-- Home level
236:             Has links down to
237:
238:             ./Daniel_Jacob/index.html
239:             ./Vacations/index.html
240:
241:         index.html generated from ./port.conf
242:         Daniel_Jacob/
243:             port.conf <-- 2nd Level
244:                 Has links up to Home Level:
245:                 ../index.html
246:
247:                 and down to:
248:
249:                 ./DJ_month_00/index.html
250:                 ./DJ_month_02/index.html
251:
252:             index.html generated from ./port.conf
253:             DJ_month_00/
254:                 index.html
255:             DJ_month_02/
256:                 index.html
257:
258:         Vacations/
259:             port.conf <-- 2nd Level
260:                 Has links up to Home Level:
261:                 ../index.html
262:
263:                 and down to:
264:
265:                 ./Honeymoon/index.html
266:                 ./Seattle/index.html
267:
268:             index.html generated from ./port.conf
269:             Honeymoon/

```

```
269:                                index.html
270:                                Seattle/
271:                                index.html
272:
273: So, at the Top or Home Level,
274:
275: portfolioNavigation:Home
276:
277: At the second level:
278:
279: portfolioNavigation:LINK([[Home]], [[../index.html]]) | BASENAME([[ $PWD ]])
```

```

1: m4_include(`./macros/basicHTMLMacros.m4')
2: m4_include(`./macros/portMacros.m4')
3: HTML_START([[Dan Mazzola]], [[Mazzola Family Photos]])
4:
5: BODY_START
6: PORTFOLIO_HEADING([[Mazzola Family Photos]], [[These are photographs taken at vari
ous family events]])
7:
8: PORTFOLIO_NAV([[Home]])
9:
10: PORTFOLIO_RECORD([[Daniel Jacob]], [[./Daniel_Jacob]], [[./Daniel_Jacob/DJ_month00/i
mages/thumb/IMG_1229.JPG]], [[See photos of our little man!]], [[12/6/2004]])
11:
12: PORTFOLIO_RECORD([[Vacations]], [[./Vacations]], [[./Vacations/Honeymoon/images/thum
b/img_0114.jpg]], [[See our Honeymoon and Seattle vacation photos]], [[12/6/2004]])
13:
14: PORTFOLIO_FOOTNOTE([[Enjoy the pictures!]])
15:
16: BODY_END
17: HTML_END
18: TABLE_END
19: HTML_END

```

```

1: #-----
2: # port.conf - skeleton file
3: #
4: # version 1.0 12/6/04
5: # version 1.1 4/28/15
6: # Dan Mazzola
7: #
8: # To be edited by hand and read by genPort (which calls awkPort)
9: # to create index.m4 and index.html which serve as portfolio
10: # summary index pages.
11: #-----
12:
13: #-----
14: # portfolioTitle: Fill in portfolio title for the web browser header bar
15: #-----
16:
17: portfolioTitle:Mazzola Family Photos
18:
19: #-----
20: # portfolioAuthor: Name of web page author
21: #-----
22:
23: portfolioAuthor:Dan Mazzola
24:
25: #-----
26: # portfolioName: Name of the current portfolio
27: #-----
28:
29: portfolioName:Mazzola Family Photos
30:
31: #-----
32: # portfolioDescription: Long text/html/macro description of this portfolio
33: #-----
34:
35: portfolioDescription:These are photographs taken at various family events
36:
37: #-----
38: # portfolioNavigation: text/html/macro string that allows navigation
39: # up or down the gallery/portfolio chain. Will be different for each
40: # level in the chain.
41: #
42: # At the Top or Home Level:
43: #
44: # portfolioNavigation:Home
45: #
46: # At the second level:
47: #
48: # portfolioNavigation:LINK([[Home]], [[../index.html]]) | BASENAME([[ $PWD]])
49: #-----
50: portfolioNavigation:Home
51:
52: #-----
53: # For each sub-portfolio or gallery, fill in the information for
54: # each field:
55: #
56: # galleryName: Name of the sub-gallery or sub-portfolio
57: # galleryDir: relative pathname of the sub-gallery or sub-portfolio
58: # galleryPhoto: relative pathname of a representative thumbnail image
59: # galleryCaption: Long text/html/macro description of this sub gallery/portfolio
60: # galleryUpdate: date of last update
61: #-----
62:
63: galleryName:Daniel Jacob
64: galleryDir:../Daniel_Jacob
65: galleryPhoto:../Daniel_Jacob/DJ_month00/images/thumb/IMG_1229.JPG
66: galleryCaption:See photos of our little man!
67: galleryUpdate:12/6/2004

```

```

68:
69: galleryName:Vacations
70: galleryDir:../Vacations
71: galleryPhoto:../Vacations/Honeymoon/images/thumb/img_0114.jpg
72: galleryCaption:See our Honeymoon and Seattle vacation photos
73: galleryUpdate:12/6/2004
74:
75: #-----
76: # portfolioFootnote - Standard footnote for all portfolio pages
77: #-----
78:
79: portfolioFootnote:Enjoy the pictures!

```

```

1: C portMacros.m4:
2: C
3: C Portfolio Macros - for use with the genPort and awkPort scripts
4: C
5: C These are macros specifically designed to support the generation of
6: C portfolios as navigational aids to get to galleries generated by
7: C genPages.
8: C
9: C Dan Mazzola
10: C version 2.0 4/28/05 - removed html tables and uses CSS and divs
11: C
12: C Make sure to include these macros in genPort.
13: C For example, to work correctly, use something like this:
14: C
15: C     echo "m4_include(\`${macrosDir}/basicHTMLMacros.m4`)"
16: C     echo "m4_include(\`${macrosDir}/port2Macros.m4`)"
17: C
18: C -----
19: C
20: C -----
21: C CHANGE QUOTE
22: C =====
23: C For this macro file, we will use
24: C "[[" (two opening square brackets) to open
25: C "]" (two closing square brackets) to close all macros
26: C Why? Two reasons, one, it is really hard for genPages to output ` and `
27: C and it will be easy to see
28: C -----
29: m4_changequote([[, ]])
30: C
31: C -----
32: C HTML_START and HTML_END
33: C =====
34: C The opening and closing macros to each page
35: C NOTE: AUTHOR, TITLE, and path to the css file is passed in as arguments
36: C -----
37: m4_define([[HTML_START]],
38: [[
39:     <!DOCTYPE html>
40:     <html>
41:     <head>
42:         <meta content='noindex, nofollow' name='robots'>
43:         <meta charset='utf-8'>
44:         <meta name="generator" content="m4">
45:         <meta name="Author" content="$1">
46:
47:         <link href="$3"
48:             rel="stylesheet" type="text/css" />
49:
50:         <title>$2</title>
51:     </head>
52: ]])
53: C
54: m4_define([[HTML_END]], [[</html>]])
55: C
56: C -----
57: C BODY_START and BODY_END
58: C =====
59: C trivially opens the closes the body
60: C -----
61: m4_define([[BODY_START]],
62: [[
63:     <body>
64:         <div id="bounding_box">
65: ]])
66: m4_define([[BODY_END]],
67: [[

```

```

68:         </div> <!-- close bounding_box -->
69:     </body>
70:
71: ]])
72: C
73: C -----
74: C PORTFOLIO_HEADING: This displays a heading for the portfolio
75: C
76: C Usage: PORTFOLIO_HEADING([[page title]], [[long text description]])
77: C -----
78: C
79: m4_define([[PORTFOLIO_HEADING]],
80: [[
81:     <div id="heading_box">
82:         <div id="main_heading">
83:             $1
84:         </div>
85:         <div id="sub_heading">
86:             $2
87:         </div>
88:     </div>
89: ]])
90: C
91: C -----
92: C PORTFOLIO_NAV: This is a simple table of LINKS for navigating up and
93: C down the portfolio
94: C
95: C Usage: PORTFOLIO_NAV([[ navigation LINKS ]])
96: C -----
97: m4_define([[PORTFOLIO_NAV]],
98: [[
99:     <div id="portfolio_navigation">
100:         $1
101:     </div>
102: ]])
103: C
104: C -----
105: C PORTFOLIO_RECORD: This is the master record for each sub portfolio or
106: C gallery
107: C
108: C Usage: PORTFOLIO_RECORD([[portfolio title]], [[./child_dir]],
109: C [[./path/to/thumbnail/image]], [[portfolio description text]],
110: C [[ date of last update ]]
111: C -----
112: m4_define([[PORTFOLIO_RECORD]],
113: [[
114:     <div class="portfolio_record_box">
115:         <div class="portfolio_record_thumbnail_box">
116:             IMAGE_LINK([[ $3 ]], [[ $2/index.html ]])
117:         </div>
118:         <div class="portfolio_record_text_box">
119:             <div class="portfolio_record_main_heading">
120:                 $1
121:             </div>
122:             <div class="portfolio_record_sub_heading">
123:                 $4
124:             </div>
125:             <div class="portfolio_record_update_stamp">
126:                 $5
127:             </div>
128:         </div>
129:     </div>
130: ]])
131: C
132: C -----
133: C PORTFOLIO_FOOTNOTE:
134: C

```

```

135: C Usage: PORTFOLIO_FOOTNOTE([[footnote text]])
136: C -----
137: m4_define([[PORTFOLIO_FOOTNOTE]],
138: [[
139:     <div id="portfolio_record_footer_box">
140:         <div id="portfolio_record_footer_text">
141:             $1
142:         </div>
143:     </div>
144: ]])
145:
146:
147: C -----
148: C IMAGE_LINK url, image_file_path
149: C =====
150: C IMAGE_LINK([[http://www.asu.edu]], [[./images/orig/123.jpg]])]
151: C -----
152: m4_define([[IMAGE_LINK]], [[<a href="$2"></a>]])
153:
154: C -----
155: C BASENAME: returns the basename of a path e.g. if path is a/b/c BASENAME
156: C calls basename(1) and returns c
157: C
158: C Usage: BASENAME(path)
159: C -----
160: m4_define([[BASENAME]],
161: [[
162:     m4_syscmd([[usr/bin/basename $1]])
163: ]])

```



```
1: #-----
2: # port.conf - skeleton file
3: #
4: # version 1.0 12/6/04
5: # version 1.1 4/28/15
6: # Dan Mazzola
7: #
8: # To be edited by hand and read by genPort (which calls awkPort)
9: # to create index.m4 and index.html which serve as portfolio
10: # summary index pages.
11: #-----
12:
13: #-----
14: # portfolioTitle: Fill in portfolio title for the web browser header bar
15: #-----
16:
17: portfolioTitle:
18:
19: #-----
20: # portfolioAuthor: Name of web page author
21: #-----
22:
23: portfolioAuthor:
24:
25: #-----
26: # portfolioName: Name of the current portfolio
27: #-----
28:
29: portfolioName:
30:
31: #-----
32: # portfolioDescription: Long text/html/macro description of this portfolio
33: #-----
34:
35: portfolioDescription:<long text description of portfolio>
36:
37: #-----
38: # portfolioNavigation: text/html/macro string that allows navigation
39: # up or down the gallery/portfolio chain. Will be different for each
40: # level in the chain.
41: #
42: # At the Top or Home Level:
43: #
44: # portfolioNavigation:Home
45: #
46: # At the second level:
47: #
48: # portfolioNavigation:LINK([[Home]], [[../index.html]]) | BASENAME([[ $PWD ]])
49: #-----
50: portfolioNavigation:
51:
52: #-----
53: # For each sub-portfolio or gallery, fill in the information for
54: # each field:
55: #
56: # galleryName: Name of the sub-gallery or sub-portfolio
57: # galleryDir: relative pathname of the sub-gallery or sub-portfolio
58: # galleryPhoto: relative pathname of a representative thumbnail image
59: # galleryCaption: Long text/html/macro description of this sub gallery/portfolio
60: # galleryUpdate: date of last update
61: #-----
62:
63: galleryName:
64: galleryDir:
65: galleryPhoto:
66: galleryCaption:
67: galleryUpdate:
```

```
68:
69: galleryName:
70: galleryDir:
71: galleryPhoto:
72: galleryCaption:
73: galleryUpdate:
74:
75: #-----
76: # portfolioFootnote - Standard footnote for all portfolio pages
77: #-----
78:
79: portfolioFootnote:
```

```

1: #!/bin/bash -p
2:
3: #-----
4: # genPort
5: #
6: # Generate a portfolio navigation page.
7: #
8: # Usage: genPort siteRootDir portConfDir
9: # where:
10: #   siteRootDir is the relative directory for the home of your website
11: #   portConfDir is the relative directory for the port.conf file
12: #
13: # /home/you/www/port2 is the absolute root
14: # /home/you/www/port2/Daniel_Jacob/ is the absolute path for the port.conf
15: # make current directory /home/you/www/port2/Daniel_Jacob and run
16: #
17: # genPort ..
18: #   will find the root of your website in the parent directory,
19: #   and the port.conf in current directory
20: #
21: # Dan Mazzola, 12/6/04, version 1.0
22: # Dan Mazzola, 04/27/15, version 1.1 convert to bash, use css
23: #-----
24:
25: # see the main function for how these variables are initialized
26: siteRootDir=""           # top level website directory
27: portConfFile="port.conf" # basename of default portfolio config file
28: portDir=""              # directory of the portfolio page
29: portConfPath=""         # path to portfolio configuration file
30: m4File="index.m4"       # basename of m4 output file
31: m4out=""                # pathname of m4 output file
32: htmlFile="index.html"   # basename of html output file
33: htmlout=""              # pathname of html output file
34: macrosDir="macros"      # dir of macro definition files
35: stylesDir="styles"      # dir of styles directory
36: cssFile="portStyles.css" # basename of the css file
37: cssPath=""              # pathname of the css file
38:
39: errorCode=0              # global error code for testing file perms
40: HTML_TIDY="/.tidyrc"     # location of tidy configuration
41:
42: #-----
43: # function startm4
44: #
45: # Called at the start of each page in function genPortPage
46: #
47: # Have this function echo items to be included in the top of each m4 file
48: #-----
49: function startm4
50: {
51:     echo "m4_include(\`${macrosDir}/basicHTMLMacros.m4`)"
52:     echo "m4_include(\`${macrosDir}/portMacros.m4`)"
53: }
54:
55: #-----
56: # function endm4
57: #
58: # Called at the end of each page in function genPortPage
59: #
60: # Have this function echo items to be included in the bottom of each
61: # m4 file
62: #-----
63: function endm4
64: {
65:     echo "BODY_END"
66:     echo "HTML_END"
67: }

```

```

68:
69: #-----
70: # genPortPage
71: #
72: #-----
73: function genPortPage
74: {
75:     # start generation of page in m4 format saved to $m4out
76:     startm4 > $m4out
77:
78:     awkPort -v cssPath=$cssPath $portConfPath >> $m4out
79:
80:     # write ending m4 code
81:     endm4 >> $m4out
82:
83:     # display page generation progress
84:     echoerr "Generating html page:\t${htmlout}"
85:
86:     # convert m4 to html and save in $htmlout
87:     m4 -P $m4out | tidy > ${htmlout}
88: }
89:
90: #-----
91: # testFile
92: #
93: # Generic file permission testing - if any test fails, generate error
94: # message and increment global errorCode value - see testDirs
95: #-----
96: function testFile
97: {
98:     scriptName=$(basename $0)
99:
100:     case $1 in
101:
102:         -d) if [ ! -d $2 ]
103:             then
104:                 echoerr "${scriptName}: Error - $2 not a directory"
105:                 let errorCode=errorCode+1
106:             fi
107:             ;;
108:
109:         -r) if [ ! -r $2 ]
110:             then
111:                 echoerr "${scriptName}: Error - $2 not readable"
112:                 let errorCode=errorCode+1
113:             fi
114:             ;;
115:
116:         -w) if [ ! -w $2 ]
117:             then
118:                 echoerr "${scriptName}: Error - $2 not writable"
119:                 let errorCode=errorCode+1
120:             fi
121:             ;;
122:
123:         -x) if [ ! -x $2 ]
124:             then
125:                 echoerr "${scriptName}: Error - $2 not searchable"
126:                 let errorCode=errorCode+1
127:             fi
128:             ;;
129:
130:         -f) if [ ! -f $2 ]
131:             then
132:                 echoerr "${scriptName}: Error - $2 not a regular file"
133:                 let errorCode=errorCode+1
134:             fi

```

```

135:             ;;
136:
137:         -e)      if [ ! -e $2 ]
138:                 then
139:                     echoerr "${scriptName}: Error - $2 does not exist"
140:                     let errorCode=errorCode+1
141:                 fi
142:             ;;
143:
144:         esac
145: }
146:
147: #-----
148: # testFiles
149: #
150: # Collects all of the tests needed to run the script. This sets a global
151: # variable errorCode to zero, then runs a sequence of tests. If any
152: # test fails, errorCode gets changed to non-zero - see testFile
153: #-----
154: function testFiles
155: {
156:     let errorCode=0
157:
158:     testFile -d ${siteRootDir}
159:     testFile -r ${siteRootDir}
160:     testFile -x ${siteRootDir}
161:     testFile -w ${siteRootDir}
162:
163:     testFile -d ${portDir}
164:     testFile -r ${portDir}
165:     testFile -x ${portDir}
166:     testFile -w ${portDir}
167:
168:     testFile -e ${portConfPath}
169:     testFile -r ${portConfPath}
170:
171:     testFile -d ${macrosDir}
172:     testFile -r ${macrosDir}
173:     testFile -x ${macrosDir}
174:
175:     testFile -d ${stylesDir}
176:     testFile -r ${stylesDir}
177:     testFile -x ${stylesDir}
178:
179:     testFile -e ${stylesPath}
180:     testFile -r ${stylesPath}
181:
182:     if [ $errorCode -gt 0 ]
183:     then
184:         echoerr "${basename $0}: ${errorCode} errors"
185:         exit 2
186:     fi
187: }
188:
189: #-----
190: # Function echoerr:
191: #
192: # This function prints messages to the standard error
193: #-----
194: function echoerr
195: {
196:     echo -e "$*" 1>&2
197: }
198:
199: #-----
200: # Function usage -
201: #

```

```

202: # Shows proper usage of the script
203: #-----
204: function usage
205: {
206:     echoerr "Usage: ${basename $0} siteRootDir portConfDir"
207:     exit 1
208: }
209:
210: #-----
211: # Main Logic
212: #-----
213: case $# in
214:     0)      usage;;
215:     1)      usage;;
216:     2)      siteRootDir=$1
217:             portDir=$2
218:             portConfPath="${portDir}/${portConfFile}"
219:             m4out="${portDir}/${m4File}"
220:             htmlout="${portDir}/${htmlFile}"
221:             macrosDir="${siteRootDir}/${macrosDir}"
222:             stylesDir="${siteRootDir}/${stylesDir}"
223:             cssPath="${stylesDir}/${cssFile}"
224:
225:             testFiles
226:             genPortPage
227:             ;;
228:
229:     *)      usage;;
230: esac

```

```

1: #!/usr/bin/awk -f
2: # Read a portfolio configuration file and generate macros based upon
3: # portMacros.m4
4: #
5: # see port.conf for file format skeleton
6: # In a nutshell, it is a file with name:value pairs, ":" is the delimiter.
7: #
8: # usage: awkport -v cssPath="path_to_css_file"
9:
10: BEGIN {
11:     FS=":"
12: }
13:
14: /^portfolioTitle/ {
15:     portTitle=$2
16:     next
17: }
18:
19: /^portfolioAuthor/ {
20:     portAuthor=$2
21:     printf("HTML_START([[%s]], [[%s]], [[%s]])\n\n",
22:         portAuthor, portTitle, cssPath)
23:     printf("BODY_START\n")
24:     next
25: }
26:
27: /^portfolioName/ {
28:     portName=$2
29:     next
30: }
31:
32: /^portfolioDescription/ {
33:     portDesc=$2
34:     printf("PORTFOLIO_HEADING([[%s]], [[%s]])\n\n", portName, portDesc)
35:     next
36: }
37: /^portfolioNavigation/ {portNav=$2
38:     printf("PORTFOLIO_NAV([[%s]])\n\n", portNav)
39:     next
40: }
41:
42: /^galleryName/ { gName=$2; next}
43: /^galleryDir/ { gDir=$2; next}
44: /^galleryPhoto/ { gPhoto=$2; next}
45: /^galleryCaption/ { gCaption=$2; next}
46: /^galleryUpdate/ {
47:     gUpdate=$2;
48:     printf("PORTFOLIO_RECORD([[%s]], [[%s]], [[%s]], [[%s]], [[%s]])\n\n",
49:         gName, gDir, gPhoto, gCaption, gUpdate)
50:     next
51: }
52:
53: /^portfolioFootnote/ {
54:     portFoot=$2
55:     printf("PORTFOLIO_FOOTNOTE([[%s]])\n\n", portFoot)
56:     next
57: }
58:

```

```

1: #!/bin/bash -p
2: #-----
3: # genNC - generate Navigation Cell data
4: #
5: # This is a sample methodology for developing the data
6: # needed to fill in the navigationCell data in genPages.
7: #
8: # The idea is this... everywhere that a port.conf file lives,
9: # in the upper layers of the portfolio, there is an index.html
10: # which serves as a navigation aid to go down the tree.
11: # Given that, we can generate the Navigation Cell data by
12: # searching up the directory heirarchy.
13: #-----
14:
15: #####
16: # customize these variables below:
17: # webHome is the top level directory for your web site
18: # portConfFile the name given to the port.conf file
19: #####
20:
21: webHome="port2"
22: portConfFile="port.conf"
23:
24: # Save the starting working directory - so we can come back!
25: startDir=$(pwd)
26:
27: # Keeps track of the current directories basename
28: curDir=""
29:
30: # used to build the URL's
31: URLup="../"
32: URLlevel=""
33:
34: # starting from the current directory (which should be a gallery dir)
35: # cd up one level and check for a port.conf file. If found, print
36: # the name of the directory and the relative URL to it
37:
38: while true
39: do
40:     cd ${URLup}
41:     URLlevel="${URLlevel}${URLup}"
42:
43:     if [[ ! -e $portConfFile ]]
44:     then
45:
46:         # we have gone one level up from our web site - gone too far
47:         # return to starting directory and break out of loop
48:
49:         cd $startDir
50:         break
51:     else
52:
53:         curDir=$(basename $(pwd))
54:
55:         # test if we are at the top of the website, and substitute
56:         # the string "Home" for the name of it
57:
58:         if [[ $curDir == $webHome ]]
59:         then
60:             curDir=Home
61:         fi
62:
63:         # Now print out the portfolio name and its URL relative
64:         # to the gallery pages.
65:
66:         echo "${curDir} is url ./${URLlevel}index.html"
67:

```

```

68:         fi
69: done

```