

```

1 #####
2 # SCRIPT:  SENTRY_STATS.PL
3 # =====
4 #
5 # FUNCTION: Perl that triggers data to be taken from a data file and stored in a table.
6 # =====
7 # PROJECT# TASK   DATE      TECH ACTION
8 # -----
9 # PLI32502.31220 19-JAN-2017 JPC  INITIAL CREATION
10 #
11 #####
12
13 # variable declarations
14 my $instance = 'PROD';
15
16 #include the error handling library
17 require Errors;
18 require Perlutil;
19
20 #set userdir from perlutil
21 my $userdir      = Perlutil->return_users_dir();
22
23
24 #Set the strings that the error logging will search for
25 Perlutil->set_search_string('JOBABEND',
26                             'WARNING',
27                             'SQLCODE',
28                             'ORA-',
29                             'SPF-',
30                             'ERROR OCCURRED:',
31                             '-- MODIFY VARIABLES AS FOLLOWS: --',
32                             '-- ADD THIS LOGIC: --',
33                             '-- JUST BEFORE THESE LINES: --',
34                             'SENTRY_STATS has completed successfully'
35                             );
36
37 my $pid = 0;
38 my $timestamp;
39 my $dos_cmd = " ";
40 my $process_error;
41 my $sqldir;
42 my $rptdir;
43 my $job_error = 'NO';
44 my $job_error_message;
45 my $progrname;
46
47 Perlutil->set_instance($instance);
48
49 require Errors;
50 require Perlutil;
51
52 # set directories
53 $sqldir = Perlutil->set_sql_dir("\\\\FLORISTS\\dfs_root\\Apps\\Sql\\");
54 $rptdir = Perlutil->set_rpt_dir($rptdir);

```

```

55
56 $programe = 'SENTRY_STATS';
57
58
59 #####
60 #                                     #
61 #                 PERL SUBMIT          #
62 #####
63
64 print "*****\n";
65 print "***** " . $programe . ":PERL_SUBMIT! *****\n";
66 print "*****\n";
67
68 $process_error = Perlutil->run_sql('SENTRY_STAT_LOADER');
69 if ($process_error eq 'YES')
70 {
71     print "JOBABEND - Errors occurred in SENTRY_STAT_LOADER \n";
72     print "May be held until next business day \n";
73     $job_error = 'WARNING';
74 }
75
76 print "dos_cmd = $dos_cmd \n";
77 system($dos_cmd);
78
79 goto ERROR_ROUTINE;
80
81
82 ###-###-###-
83
84 $process_error = Perlutil->run_sql('SENTRY_STATS_VALIDATION');
85 if ($process_error eq 'YES')
86 {
87     print "JOBABEND - Errors occurred in SENTRY_STATS_VALIDATION \n";
88     print "May be held until next business day \n";
89     $job_error = 'WARNING';
90 }
91
92 print "dos_cmd = $dos_cmd \n";
93 system($dos_cmd);
94
95 goto ERROR_ROUTINE;
96
97
98 ###-###-###-
99
100 $process_error = Perlutil->run_sql('SENTRY_STATS_XFER');
101 if ($process_error eq 'YES')
102 {
103     print "JOBABEND - Errors occurred in SENTRY_STATS \n";
104     print "May be held until next business day \n";
105     $job_error = 'WARNING';
106 }
107
108 print "dos_cmd = $dos_cmd \n";

```

```

109 system($dos_cmd);
110
111 goto ERROR_ROUTINE;
112
113
114 ###--
115
116 #####
117 # ERROR_ROUTINE: ROUTINE FOR FATAL ERRORS ENCOUNTERED #
118 #####
119 ERROR_ROUTINE:
120 if ($job_error eq 'FATAL')
121 {
122     print "*!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!*\n";
123     print "*"                                           \n";
124     print " JOBABEND - Fatal Errors occurred in " . $programe . " - contact programming!!!   \n";
125     print "$job_error_message                                \n";
126     print "*"                                           \n";
127     print "*!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!*\n";
128
129 Perlutil->insert_array_value('*****');
130 ***);
131
132 Perlutil->insert_array_value('*****');
133 ***);
134 Perlutil->insert_array_value('JOBABEND - Fatal Errors occurred in ' . $programe . "' - contact
programming ');
Perlutil->insert_array_value($job_error_message);
135
Perlutil->insert_array_value('*****');
136 ***);
137
# create, email and print an error log file
Perlutil->search_log($error_logfile);
die;
140 }
141
#####
142 # DISPLAY COMPLETED MESSAGE #
143 #####
144
145 # check for warnings
146 if ($job_error eq 'WARNING')
147 {
148     print "*!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!*\n";
149     print "*"                                           \n";
150     print "< WARNING > - non-fatal errors occured in SENTRY_STATS.PL\n";
151     print "      " . $programe . " did complete, but check log file for instructions    \n";
152     print "*"                                           \n";
153     print "*!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!*\n";

```

```

155
156 Perlutil->insert_array_value('*****');
157 Perlutil->insert_array_value('*****');
158 Perlutil->insert_array_value(' < WARNING > - non-fatal errors occurred in ' . $programe . " ");
159 Perlutil->insert_array_value(' ' . $programe . ' has completed to end-of-job, ');
160 Perlutil->insert_array_value(' but check log file for instructions' );
161 Perlutil->insert_array_value('*****');
162 Perlutil->insert_array_value('*****');
163 }
164 else
165 {
166     print "*****\n";
167     print "*" \n";
168     print " " . $programe . " has completed successfully \n";
169     print "*" \n";
170     print "*****\n";
171
172     Perlutil->insert_array_value('*****');
173     Perlutil->insert_array_value('*****');
174     Perlutil->insert_array_value(' SENTRY_STATS.PL has completed successfully');
175     Perlutil->insert_array_value('*****');
176     Perlutil->insert_array_value('*****');
177 }
178
179 # warning or success: create, email and print an error log file
180 Perlutil->search_log($error_logfile);
181 exit;
182

```