

```
1 <html>
2 <head>
3   <script type="text/javascript" src="https://code.jquery.com/jquery-2.2.3.js"></script>
4 </head>
5 <body>
6
7   <?php
8     # error_reporting(E_ALL);
9
10    //ini_set('display_errors', TRUE) ;
11    //error_reporting(E_ALL) ;
12
13    global $link ;
14    global $show, $silent ;
15    global $pathToHome, $urlToHome ;
16
17    $show = false ;           // Show DATA if true
18    $silent = false ;        // Show progress messages if false
19
20    $pathToHome="C:/XAMPP/htdocs/cmdb/" ;
21    $urlToHome="/cmdb/" ;
22    set_include_path(get_include_path() . PATH_SEPARATOR .
23      'C:/Users/smithr2/Documents/aa My STUFF/b Cell Modem Management/Source
24      Material/Sources, Cleaned');
25
26    if ( php_uname('n') == 'a-hostname' )
27    {
28      $pathToHome="/home/ralph/cmdb/" ;
29      $urlToHome="http://a-hostname/cmdb/" ;
30      set_include_path(get_include_path() . PATH_SEPARATOR .
31        '/home/ralph/import_source' ) ;
32    }
33
34
35    require_once $pathToHome.'PHP/import_Library.php';
36    require_once $pathToHome.'PHP/library.php';
37
38    date_default_timezone_set("America/Los_Angeles") ;
39
40    clearstatcache(true) ;
41
42
43    #
44    =====
45    =====
46
47    # LOAD HTML BELOW
48
49    $content =
50      file_get_contents($pathToHome."public_html/Import/import_CellModemPlans_ATnT.html")
```

```
51     ,true) ;
52     echo($content) ;
53
54     # HTML ABOVE
55     #
56     =====
57     =====
58     # Show status
59
60     echo "
61         <script type='text/javascript' >
62             $('#import_CellModemPlans_message1').empty() ;
63             $('#import_CellModemPlans_message1').html('Form submission message status
64                 box.')
```

```
101     $file_destination0=$pathToHome.'../import_source/'.basename(  
102         $_FILES['import_file_path']['name'] ) ;  
103     $file_destination = str_replace(" ", "_", $file_destination0) ;  
104  
105     if ( move_uploaded_file($_FILES['import_file_path']['tmp_name'],  
106         $file_destination) )  
107     {  
108         echo "<p>The file ". basename( $_FILES['import_file_path']['name'] )." has been  
109             uploaded successfully.</p>";  
110     }  
111     else  
112     {  
113         echo "<p>There was an error uploading the file, please try again!</p>";  
114         exit() ;  
115     }  
116  
117     $whatA = clean_form_fields($_POST["import_month"]) ;  
118     $whatB = clean_form_fields($_POST["import_year"]) ;  
119  
120     $posting_period = $whatA." ".$whatB;                // MMM YYYY  
121     $invoice_type="Modem Plans for current period" ;  
122     $invoice_identifier = $posting_period." ".$invoice_type ; // MMM YYYY Modem Plans  
123         for current period  
124  
125     $who = clean_form_fields($_POST["who"]) ;  
126     $who=clean_punctuation($who) ;  
127  
128     $description = clean_form_fields($_POST["description"]) ;  
129     $description=clean_punctuation($description) ;  
130  
131  
132     $mykeys = array('wireless_no', 'subscription_status', 'date_latest_active_began',  
133         'domplan_code', 'domplan_name', 'domplan_rate', 'intlplan_code',  
134         'intlplan_name', 'intlplan_rate') ;  
135  
136  
137  
138     //  
139     -----  
140     -----  
141     if (strpos($Err, 'Please watch your slash characters.'))  
142     {  
143         echo "  
144             <script type='text/javascript' >         alert('slashes slashes slashes!');  
145                 $('#import_CellModemPlans_message1').empty() ;  
146                 $('#import_CellModemPlans_message1').html('Watch your slashes!') ;  
147             </script>" ;  
148         exit();  
149     }  
150     elseif (strpos($Err, "The file ".$filename." does not exist."))
```

```
151 {
152     echo "
153         <script type='text/javascript' >         alert('File not found. Please try
154             again. ');
155             $('#import_CellModemPlans_message1').empty() ;
156             $('#import_CellModemPlans_message1').html('Please check the file name!') ;
157         </script>" ;
158     exit();
159 }
160
161 // =====
162 // =====
163 list($usec, $sec) = explode(' ', microtime());
164 $script_start = (float) $sec + (float) $usec;
165 echo "script_start (UNIX seconds since 1970-01-01) = ".$script_start ;
166 // =====
167
168
169 readfile_into_strarray($file_destination,1,$nlines, $array_of_lines) ;
170
171 //if ($show) { echo "<p><pre>"; print_r($array_of_lines) ; echo "</pre><p>" ; }
172
173 // =====
174 // Remove quotes and dollar signs
175 // Replace / with - in the process of changing MM/DD/YYYY to YYYY-MM-DD
176 // Lastly return with the true # of non-empty lines
177
178 if (!$silent) { echo "<p>Entering function array_clean_ReplRemvDatefix()" ; }
179
180 array_clean_ReplRemvDatefix($nlines, $array_of_lines, "\",\$", "/", "-",
181     $cleaned_lines) ;
182
183 if (!$silent) { echo "<br>Returned from function array_clean_ReplRemvDatefix()" ;
184     }
185
186 // The above also returned the # of lines that actually have data, so the cmd
187 // below
188 // takes all of the rows up to that many,
189 // GREATLY ASSUMING the empties were at the end!
190
191 $cleaned_lines=array_slice ($cleaned_lines, 0, $nlines) ;
192
193 if ($show) { echo "<p><pre>"; print_r($cleaned_lines) ; echo "</pre><p>" ; }
194
195 // =====
196 # MUST DO THIS BEFORE YOU USE mysqli_real_escape_string BECAUSE IT NEEDS THE LINK
197 #
198 # Get a DB connection with cooler options (besides those explicitly set
199     (successfully))
200
```

```
201     if (!$silent) { echo "<p>Entering function connect_to_mysql()" ; }
202     $link = connect_to_mysql() ;
203
204     if (!$silent) { echo "<br>Returned from function connect_to_mysql()" ; }
205
206     # MUST DO THIS BEFORE YOU USE mysqli_real_escape_string BECAUSE IT NEEDS THE LINK
207     #
208
209     // =====
210     # change from a 1D array of strings to a 2D array of rows and columns
211     # This uses mysqli_real_escape_string, so you need the $link first (above).
212
213     if (!$silent) { echo "<p>Entering function str_array_1DwTabs_to_2D()" ; }
214
215     str_array_1DwTabs_to_2D($nlines, $cleaned_lines, $mykeys, $desired_data) ;
216
217     if (!$silent) { echo "<br>Returned from function str_array_1DwTabs_to_2D()" ; }
218     if ($show) { echo "<p><pre>" ; print_r($desired_data) ; echo "</pre><p>" ; }
219
220     // =====
221     // TRIM every row's columns
222
223     if (!$silent) { echo "<p>Entering function trim_every_cell()" ; }
224
225     trim_every_cell($mykeys, $nlines, $desired_data) ;
226
227     if (!$silent) { echo "<br>Returned from function trim_every_cell()" ; }
228
229
230     // =====
231     // TRIM every row's columns
232
233     if (!$silent) { echo "<p>Entering function cannot_be_nulls()" ; }
234
235     $gotNullsError=0 ;
236     $colsToCheckWithAND = array() ;
237     $colsToCheckWithAND[] = 'wireless_no' ;
238
239     cannot_be_nulls($gotNullsError, $colsToCheckWithAND, $nlines, $desired_data) ;
240
241     if ( $gotNullsError === 1 )
242     {
243         echo "<p><span style='color:red'><strong>Because of the date errors mentioned
244             above this program has stopped.</strong></span></p>" ;
245         exit() ;
246     }
247
248     if (!$silent) { echo "<br>Returned from function cannot_be_nulls()" ; }
249
250     // =====
```

```
251 // Check any fields that contain a date for conformity
252
253 if (!$silent) { echo "<p>Entering function check_date_conformity()" ; }
254
255 $dateError = 0 ;
256 check_date_conformity($dateError, $mykeys, $nlines, $desired_data) ;
257
258 if ( $dateError )
259 {
260     echo "<p><span style='color:red'><strong>Because of the date errors mentioned
261         above this program has stopped.</strong></span></p>" ;
262     exit() ;
263 }
264
265 if (!$silent) { echo "<br>Returned from function check_date_conformity()" ; }
266
267 // =====
268 // Check any fields that contain a wireless_no for conformity
269
270 if (!$silent) { echo "<p>Entering function check_wireless_no_conformity()" ; }
271
272 $wirelessNoError=0 ;
273 check_wireless_no_conformity($wirelessNoError, $mykeys, $nlines, $desired_data) ;
274
275 if ( $wirelessNoError === 1 )
276 {
277     echo "<p><span style='color:red'><strong>Because of the wireless_no errors
278         mentioned above this program has stopped.</strong></span></p>" ;
279     exit() ;
280 }
281
282 if (!$silent) { echo "<br>Returned from function check_wireless_no_conformity()" ; }
283 }
284
285
286 // =====
287 // Use PHP functions to just grab the columns we need, assuming they're in the
288 // order
289 // as laid out in the array $mykeys.
290
291 // Changed the requirement to have data already ordered for this import, so less
292 // to do here.
293
294 # $array_1D=array() ;
295 #
296 # for ($i=0; $i<$nlines; $i++)
297 # {
298 #     $array_1D=array_merge( array_slice($desired_data[$i], 0,8),
299 #         array_slice($desired_data[$i], 9,2), array_slice($desired_data[$i],13,2) ) ;
300 # }
```

```
301     #   $desired_data[$i]= $array_1D ;
302     # }
303
304     if ($show) { echo "<p><pre>"; print_r($desired_data) ; echo "</pre><p>" ; }
305
306
307     // =====
308     // Does this invoice exist?
309     // If necessary, assert this invoice_type, service_provider, & posting_period
310
311     if (!$silent) { echo "<p>Entering function assert_invoice() which returns with an
312         Invoice ID." ; }
313
314     $invoice_id = assert_invoice( $link, $posting_period, $invoice_type, "AT&T" ) ;
315
316     list($usec, $sec) = explode(' ', microtime());
317     $script_interim = (float) $sec + (float) $usec;
318     if (!$silent) { echo "<br>Returned from function assert_invoice(). Total elapsed
319         time ".round($script_interim - $script_start, 5)." seconds." ; }
320
321
322     // =====
323
324     if (!$silent) { echo "<p>Deleting data from 'raw_ModemPlans_ATnT'" ; }
325
326     if ( !$result = mysqli_query($link,"DELETE FROM `raw_ModemPlans_ATnT` ;") )
327     { echo "<br><span style='color:red'><strong>Error: Could not find or delete from
328         Table `raw_ModemPlans_ATnT`.</strong></span></p>" ; }
329     else
330     { echo "<br><span style='color:green;'>Table emptied.</span>" ; }
331
332     mysqli_query($link,"COMMIT ;") ;
333
334     if (!$silent) { echo "<br>Deleting completed successfully." ; }
335     if ($show) { echo "DELETED from table okay." ; }
336
337     // =====
338     # Insert the data into an all-VARCHAR table
339     # (Note that I put the necessary COMMIT in the functions/procedures)
340
341     if (!$silent) { echo "<p>Entering function SQL_insert_2D_arraydata()." ; }
342
343     SQL_insert_2D_arraydata("raw_ModemPlans_ATnT",$desired_data,9,1) ;
344
345     list($usec, $sec) = explode(' ', microtime());
346     $script_interim = (float) $sec + (float) $usec;
347     if (!$silent) { echo "<p>Returned from function SQL_insert_2D_arraydata(). Total
348         elapsed time ".round($script_interim - $script_start, 5)." seconds." ; }
349
350     // =====
```

```

351
352     if (!$silent) { echo "<p>Entering function SQL_LogImport()." ; }
353
354     SQL_LogImport($link, $who, $invoice_identifier, $file_destination0, $description)
355         ;
356
357     list($usec, $sec) = explode(' ', microtime());
358     $script_interim = (float) $sec + (float) $usec;
359     if (!$silent) { echo "<br>Returned from function SQL_LogImport(). Total elapsed
360         time ".round($script_interim - $script_start, 5)." seconds." ; }
361
362
363     // =====
364     # See how many records got imported before the browser stops things.
365
366     if (!$silent) { echo "<p>Entering function done_record_count()" ; }
367
368     done_record_count($link) ;
369
370     list($usec, $sec) = explode(' ', microtime());
371     $script_interim = (float) $sec + (float) $usec;
372     if (!$silent) { echo "<br>Returned from function done_record_count(). Total
373         elapsed time ".round($script_interim - $script_start, 5)." seconds." ; }
374
375
376     // =====
377     // Within SQL PROCEDURE ImportCellModemPlans_ATnT will
378     // 1) update Invoice.date_last_import,
379     // 2) with each wireless_no will insert into invoiced_modem that & the invoice_id
380         just attained,
381     // 3rd, will add this raw file to zArchived_Raw_Modem_Plans, (w the
382         invoice_identifier), and
383     // 4th, be sure a note exists for each new modem
384     // 5th, and a boatload more.
385
386     if (!$silent)
387     {   echo
388         "<p>=====<br>"
389         ;
390         echo "Creating a MySQL EVENT to perform the post processing on this data." ; }
391
392     $event = 'CREATE EVENT PostProcessModemPlans ' ;
393     $event = $event.'ON SCHEDULE AT CURRENT_TIMESTAMP() + INTERVAL 5 SECOND ' ;
394     $event = $event.'DO ' ;
395     $event = $event."CALL ImportCellModemPlans_ATnT('.$invoice_id.",
396         ' ".$invoice_identifier."', 1) ;" ;
397
398     if ( !$dummy = mysqli_query( $link, $event ) )
399     {   echo "<p><span style='color:red;'>CALL to ImportCellModemPlans_ATnT.sql failed!
400         ".mysqli_error($link)."</span><p>" ; }

```



```
401     else
402     { echo "<p><span style='color:green;'>EVENT created. See definition
403       below.</span><p>" ; }
404 #echo "<p>&nbsp;<p>CALL ImportCellModemPlans_ATnT(".$invoice_id.",
405       "$invoice_idenfier.", 1) ;</p></p>" ;
406
407     if (!$silent) { echo "<br>EVENT created. See definition below.<p>" ; }
408
409     // =====
410
411     sleep(2) ;
412
413     $query = "SELECT EVENT_NAME, EVENT_DEFINITION, EVENT_TYPE, EXECUTE_AT FROM
414       information_schema.EVENTS WHERE EVENT_NAME = 'PostProcessModemPlans' ORDER BY
415       LAST_ALTERED DESC LIMIT 1 ; " ;
416
417     if (!$result = mysqli_query($link, $query))
418     {
419       printf("Error: %s\n", "<p>".mysqli_error($link)." Could not query status of
420         CREATE EVENT that will run the post-processing.</p>");
421     }
422     $row = mysqli_fetch_assoc($result) ;
423     mysqli_free_result($result) ;
424
425     # -----
426     echo "<PRE>" ;
427     echo "An automatic post-processing event, described below, has been created.\n" ;
428     echo "It will log its post-processing progress into Table 'Import_Details'.\n" ;
429     echo "Event Name:      ".$row["EVENT_NAME"]."\n" ;
430     echo "Event Type:      ".$row["EVENT_TYPE"]."\n" ;
431     echo "Event Definition: ".$row["EVENT_DEFINITION"]."\n" ;
432     echo "Event Start:     ".$row["EXECUTE_AT"]."\n\n" ;
433     echo "</PRE>" ;
434
435     if (!$silent)
436     {
437       echo "<br>Post processing of this import by SQL procedure
438         'ImportCellModemPlans_ATnT' will kick off in 5 seconds." ;
439       echo "<p><span style='color:darkorange;'>Though you can run this import again
440         without complications,</span>" ;
441       echo "<br><span style='color:red;'>PLEASE WAIT until import post-processing has
442         completed before modifying the ModemConfig table or ANY NOTES.</p></span>" ;
443     }
444
445     if (!$silent) { echo "\n<p><span style='color:green;'><strong>Script
446       Import_CellModemPlans_ATnT.php has completed
447       successfully.</strong></span></p>\n" ; }
448
449     list($usec, $sec) = explode(' ', microtime());
450     $script_interim = (float) $sec + (float) $usec;
```

```
451     echo "<p>Total elapsed time: ".round($script_interim - $script_start, 5)."  
452         seconds.</p>" ;  
453     echo  
454     "===== "  
455     "===== " ;  
456  
457  
458  
459     exit() ;  
460  
461  
462  
463 } // EO Main Procedure  
464  
465 # *****  
466 # *****  
467 # MAIN program above, procedures/functions below  
468 # *****  
469 # *****  
470  
471 # *****  
472 # *****  
473  
474 function done_record_count($link)  
475 {  
476  
477     if ($result = mysqli_query($link, "SELECT COUNT(*) FROM raw_ModemPlans_ATnT;"))  
478     {  
479         $row = mysqli_fetch_row($result) ;  
480         printf("</br><span style='color:green'>Confirmed. Table 'raw_ModemPlans_ATnT'  
481             has %d rows.</span>\n", $row[0] ) ;  
482         mysqli_query($link, "COMMIT ;" ) ;  
483         mysqli_free_result($result) ;  
484     }  
485     else  
486     {  
487         echo "<span style='color:red'><strong>Error: SELECT COUNT(*) FROM  
488             raw_ModemPlans_ATnT FAILED.</strong></span></p>" ;  
489     }  
490  
491 }  
492  
493 # *****  
494 # *****  
495  
496 # *****  
497 # *****  
498  
499 function assert_invoice( $link, $posting_period, $invoice_type, $svc_provider_name )  
500 {
```

```
501      # $posting_period like "Mar 2016",    $svc_provider_name like "AT&T"
502      # $invoice_type like [ Modem charge for prior period | Modem plans for current
503      period ]
504
505      $query =          "SELECT invoice_id, posting_period, service_provider_id, " ;
506      $query = $query."DATE_FORMAT(date_plans_import_started, '%a %b %d %Y @ %T'),
507      DATE_FORMAT(date_plans_import_completed, '%a %b %d %Y @ %T') " ;
508      $query = $query."FROM    Invoice i JOIN ServiceProvider sp
509      USING(service_provider_id) " ;
510      $query = $query."WHERE    posting_period = '". $posting_period.'" " ;
511      $query = $query."    AND    sp.carrier_name = '". $svc_provider_name.'" ;" ;
512
513      if (!$result = mysqli_query($link, $query))
514      {
515          printf("Error: %s\n", "<p>".mysqli_error($link). " in function assert_invoice()
516              at first query.</p>");
517          exit() ;
518      }
519
520      $row = mysqli_fetch_row($result) ;
521      mysqli_free_result($result) ;
522
523      $invoice_id = $row[0] ;
524
525      if (is_null($invoice_id))
526      {
527          echo '\n<br><span style="color:magenta;">'."Posting Period
528              '". $posting_period.'" and Invoice Type '". $invoice_type.'" does not yet
529              exist.</span>\n' ;
530
531          # First we have to get the service_provider_id
532          $query =          "SELECT service_provider_id FROM ServiceProvider WHERE
533              carrier_name = '". $svc_provider_name.'" ;" ;
534          if (!$result = mysqli_query($link, $query))
535          {
536              printf("Error: %s\n", "<p>".mysqli_error($link). " in function assert_invoice()
537                  at second query.</p>");
538          }
539          $row = mysqli_fetch_row($result) ;
540          mysqli_free_result($result) ;
541
542          $service_provider_id = $row[0] ;
543
544          $pp_sortable =
545              substr($posting_period,4).'-'.get_2digit_numeric_month(substr($posting_period,
546              0,3)).'-00' ;
547
548          # Now to do the insert of current data
549          $query =          "INSERT INTO Invoice (posting_period          , pp_sortable          ,
550              service_provider_id          ,    date_plans_import_started) " ;
```

```
551     $query = $query."          VALUES ('".$posting_period."', '".$pp_sortable."',
552         ".$service_provider_id."', '".date("Y-m-d H:i:s")."' ) ;" ;
553
554     if (!$result = mysqli_query($link, $query))
555     {
556         printf("Error: %s\n", "<p>".mysqli_error($link). " in function assert_invoice()
557             at INSERT into Invoice statement.</p>");
558     }
559     mysqli_query($link, "COMMIT ;") ;
560
561     echo "\n<br><span style='color:magenta;'>". "Invoice with Posting Period
562         '".$posting_period."' now exists.</span>\n" ;
563
564     #-----
565     # Now to get the new ID of the Invoice table for this data
566     $query = "SELECT invoice_id " ;
567     $query = $query."FROM Invoice " ;
568     $query = $query."WHERE posting_period = '".$posting_period."' " ;
569     $query = $query." AND service_provider_id = ".$service_provider_id." ;" ;
570
571     if (!$result = mysqli_query($link, $query))
572     {
573         printf("Error: %s\n", "<p>".mysqli_error($link). " in function assert_invoice()
574             at 2nd query.</p>");
575     }
576
577     $row = mysqli_fetch_row($result) ;
578     $invoice_id = $row[0] ;
579
580     mysqli_query($link, "COMMIT ;") ;
581     mysqli_free_result($result) ;
582
583     echo "\n<br><span style='color:magenta;'>Invoice ID =
584         ".$invoice_id."</span>\n" ;
585
586     # - - - - -
587     - - - - -
588     $query = 'INSERT INTO Note (`note_src_table`, `note_src_table_id`) VALUES
589         ("Invoice", '.$invoice_id.');" ;
590     if (!$result = mysqli_query($link, $query))
591     {
592         printf("Error: %s\n", "<p>".mysqli_error($link). " in function assert_invoice()
593             at INSERT into Note statement.</p>");
594     }
595     mysqli_query($link, "COMMIT ;") ;
596     # - - - - -
597     - - - - -
598     $query = 'SELECT LAST_INSERT_ID() ;' ;
599     if (!$result = mysqli_query($link, $query))
600     {
```

```
601         printf("Error: %s\n", "<p>".mysqli_error($link)." in function assert_invoice()  
602             at INSERT into Note statement.</p>");  
603     }  
604     $row = mysqli_fetch_row($result) ;  
605     $note_row_id = $row[0] ;  
606     mysqli_free_result($result) ;  
607     # -----  
608     -----  
609     $query = 'UPDATE Invoice SET note_id = '.$note_row_id.' WHERE invoice_id =  
610         '.$invoice_id.' ' ;  
611     if (!$result = mysqli_query($link, $query))  
612     {  
613         printf("Error: %s\n", "<p>".mysqli_error($link)." in function assert_invoice()  
614             at UPDATE Invoice statement.</p>");  
615     }  
616     mysqli_query($link, "COMMIT ;") ;  
617  
618  
619 }     # -----  
620 else  # New invoice above // Existing invoice below  
621  
622 {  
623  
624     echo '\n<br><span style="color:magenta;">Modem Plans Invoice ID='.$invoice_id."  
625         for Posting Period ' ".$posting_period.'" already exists.</span>\n" ;  
626  
627     $posting_period=$row[1] ;  
628     $date_plans_import_started=$row[3] ;  
629     $date_plans_import_completed=$row[4] ;  
630  
631     if (! is_null($date_plans_import_started))  
632     {  
633         echo '<p><div style="color:red;"> NOTE:'. "\n" ;  
634         echo "<br>\nThis Plans Invoice for ' ".$posting_period.'" was previously  
635             started on ' .$date_plans_import_started.'.\n" ;  
636         if (is_null($date_plans_import_completed))  
637         {  
638             echo "<br>It could still be running up to 15 minutes later, though there is  
639                 NO COMPLETION noted as of now.\n" ;  
640         }  
641         else  
642         {  
643             echo "<br>The database shows that it was also completed on  
644                 ' .$date_plans_import_completed.'.\n" ;  
645             echo "<br>and completion assertion is the very last post-import processing  
646                 action taken.\n" ;  
647         }  
648         echo "<br>It is safe however to repeat a 'Plans' import.\n" ;  
649         echo "</div>" ;  
650     }
```

```
651
652     $query =          "UPDATE Invoice SET `date_plans_import_started`    =
653         '".date("Y-m-d H:i:s")."' ;
654     $query = $query." WHERE  invoice_id = ".$invoice_id." ; " ;
655
656     if (!$result = mysqli_query($link, $query))
657     {
658         printf("Error: %s\n", "<p>".mysqli_error($link). " in function assert_invoice()
659             at UPDATE asserting `date_plans_import_started`.</p>");
660     }
661     mysqli_query($link, "COMMIT ;") ;
662
663     } # New or existing invoice
664
665
666     return $invoice_id ;
667
668 }
669
670 ?>
671 </body>
672 </html>
673
674
```