simple_tables.pql
by *Pequel*

sample@youraddress.com

Simple Tables Example Script

2.2

Table of Contents
Simple Tables Example Script

SCRIPT NAME

simple_tables.pql

DESCRIPTION

Demonstrates the use of tables.

1. PROCESS DETAILS

Input records are read from standard input. The input record contains ***8*** fields. Fields are delimited by the '**|**' character.

Output records are written to standard output. The output record contains ***6*** fields. Fields are delimited by the '**|**' character.

1.1 *LOCATION*

Output Field

Description

Set to input field ***LOCATION***

1.2 *CITY_NAME*

Output Field

Description

Set to input field ***CITY_NAME***

Derived Input Field Evaluation

```
=> %TCITY(LOCATION)->1
```

1.3 *STATE_NAME*

Output Field

Description

Set to input field ***STATE_NAME***

Derived Input Field Evaluation

```
=> %TSTATE(%TCITY(LOCATION)->2)->1 || %TSTATE(LOCATION)->1
```

1.4 *STATE_NAME_UC*

Output Field

Description

Set to input field ***STATE_NAME_UC***

Derived Input Field Evaluation

```
=> &uc(%TSTATE(&uc(%TCITY(&uc(LOCATION))->2))->1 || %TSTATE(LOCATION)->1)
```

1.5 *STATE_CODE*

Output Field

Description

Set to input field ***STATE_CODE***

Derived Input Field Evaluation

```
=> %TCITY(&uc(LOCATION))->2 || LOCATION
```

1.6 *STATE_NAME_3*

Output Field

Description

Set to input field ***STATE_NAME_3***

Derived Input Field Evaluation

```
=> %TSTATE(STATE_CODE)->1
```

2. CONFIGURATION SETTINGS

2.1 *pequeldoc*

generate pod / pdf pequel script Reference Guide.: pdf

2.2 *detail*

Include Pequel Generated Program chapter in Pequeldoc: 1

2.3 *script_name*

script filename: simple_tables.pql

2.4 *header*

write header record to output.: 1

2.5 *optimize*

optimize generated code.: 1

2.6 *discard_header*

Input file has header record - must be discarded.: 1

2.7 *doc_title*

document title.: Simple Tables Example Script

2.8 *doc_email*

document email entry.: sample@youraddress.com

2.9 *doc_version*

document version for pequel script.: 2.2

3. TABLES

3.1 *TCITY*

Table Type: ***local***

Data

SYD — Sydney NSW
MEL — Melbourne VIC
PER — Perth WA
ALIC — Alice Springs NT

3.2 *TSTATE*

Table Type: ***local***

Data

WA — Western Australia
NSW — New South Wales
SA — South Australia
QLD — Queensland
NT — Northern Territory
VIC — Victoria

4. TABLE INFORMATION SUMMARY

4.1 Table List Sorted By Table Name

TCITY — ***1*** (*local*)
TSTATE — ***2*** (*local*)

5. SIMPLE_TABLES.PQL

options

```
pequeldoc(pdf)
detail(1)
script_name(simple_tables.pql)
header(1)
optimize(1)
discard_header(1)
doc_title(Simple Tables Example Script)
doc_email(sample@youraddress.com)
doc_version(2.2)
```

description

```
Demonstrates the use of tables.
```

init table

```
TCITY SYD Sydney NSW
TCITY MEL Melbourne VIC
TCITY PER Perth WA
TCITY ALIC Alice Springs NT

TSTATE WA Western Australia
TSTATE NSW New South Wales
TSTATE SA South Australia
TSTATE QLD Queensland
TSTATE NT Northern Territory
TSTATE VIC Victoria
```

input section

```
PRODUCT_CODE
COST_PRICE
DESCRIPTION
SALES_CODE
SALES_PRICE
SALES_QTY
SALES_DATE
LOCATION
CITY_NAME => %TCITY(LOCATION)->1

STATE_NAME => %TSTATE(%TCITY(LOCATION)->2)->1 || %TSTATE(LOCATION)->1

STATE_NAME_UC => &uc(%TSTATE(&uc(%TCITY(&uc(LOCATION))->2))->1 || %TSTATE(LOCATION)->1)

STATE_CODE => %TCITY(&uc(LOCATION))->2 || LOCATION

STATE_NAME_3 => %TSTATE(STATE_CODE)->1
```

output section

```
string    LOCATION      LOCATION
string    CITY_NAME     CITY_NAME
string    STATE_NAME    STATE_NAME
string    STATE_NAME_UC STATE_NAME_UC
string    STATE_CODE    STATE_CODE
string    STATE_NAME_3  STATE_NAME_3
```

6. PEQUEL GENERATED PROGRAM

```
# vim: syntax=perl ts=4 sw=4
#-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
#Generated By: pequel Version 2.2-9, Build: Tuesday September 13 08:43:08 BST 2005
#            : https://sourceforge.net/projects/pequel/
#Script Name : simple_tables.pql
#Created On  : Tue Sep 13 10:32:39 2005
#For         :
#-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
#Options:
#pequeldoc(pdf) generate pod / pdf pequel script Reference Guide.
#detail(1) Include Pequel Generated Program chapter in Pequeldoc
#script_name(simple_tables.pql) script filename
#header(1) write header record to output.
#optimize(1) optimize generated code.
#discard_header(1) Input file has header record - must be discarded.
#doc_title(Simple Tables Example Script) document title.
#doc_email(sample@youraddress.com) document email entry.
#doc_version(2.2) document version for pequel script.
#-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
use strict;
local $\="\n"; local $,="|";
print STDERR '[simple_tables.pql ' . localtime() . "] Init";
use constant VERBOSE => int 10000;
use constant LAST_ICELL => int 12;
my @I_VAL;
my @O_VAL;
foreach my $f (1..6) { $O_VAL[$f] = undef; }
my $_TABLE_TCITY = &InitLookupTCITY; # ref to %$TCITY hash
my $_TABLE_TSTATE = &InitLookupTSTATE; # ref to %$TSTATE hash
use constant _I_PRODUCT_CODE     => int    0;
use constant _I_COST_PRICE       => int    1;
use constant _I_DESCRIPTION      => int    2;
use constant _I_SALES_CODE       => int    3;
use constant _I_SALES_PRICE      => int    4;
use constant _I_SALES_QTY        => int    5;
use constant _I_SALES_DATE       => int    6;
use constant _I_LOCATION         => int    7;
use constant _I_CITY_NAME        => int    8;
use constant _I_STATE_NAME       => int    9;
use constant _I_STATE_NAME_UC    => int   10;
use constant _I_STATE_CODE       => int   11;
use constant _I_STATE_NAME_3     => int   12;
use constant _O_LOCATION         => int    1;
use constant _O_CITY_NAME        => int    2;
use constant _O_STATE_NAME       => int    3;
use constant _O_STATE_NAME_UC    => int    4;
use constant _O_STATE_CODE       => int    5;
use constant _O_STATE_NAME_3     => int    6;
use constant _T_TCITY_FLD_1    => int    0;
use constant _T_TCITY_FLD_2    => int    1;
use constant _T_TSTATE_FLD_1   => int    0;
use constant _I_TCITY_LOCATION_FLD_KEY   => int   13;
use constant _I_TCITY_LOCATION_FLD_1     => int   14;
use constant _I_TCITY_LOCATION_FLD_2     => int   15;
use constant _I_TCITY_1_FLD_KEY          => int   16;
use constant _I_TCITY_1_FLD_1            => int   17;
use constant _I_TCITY_1_FLD_2            => int   18;
use constant _I_TSTATE_1_FLD_KEY         => int   19;
use constant _I_TSTATE_1_FLD_1           => int   20;
use constant _I_TSTATE_LOCATION_FLD_KEY  => int   21;
use constant _I_TSTATE_LOCATION_FLD_1    => int   22;
use constant _I_TSTATE_2_FLD_KEY         => int   23;
use constant _I_TSTATE_2_FLD_1           => int   24;
use constant _I_TSTATE_STATE_CODE_FLD_KEY => int   25;
use constant _I_TSTATE_STATE_CODE_FLD_1  => int   26;
&PrintHeader();
my $discard_header = <STDIN>;
print STDERR '[simple_tables.pql ' . localtime() . "] Start";
use Benchmark;
my $benchmark_start = new Benchmark;
while (<STDIN>)
{
    print STDERR '[simple_tables.pql ' . localtime() . "] $. records." if ($. % VERBOSE == 0);
    chomp;
    @I_VAL = split("[|]", $_);
    $O_VAL[_O_LOCATION] = $I_VAL[_I_LOCATION];
    $I_VAL[_I_CITY_NAME] = ${$$_TABLE_TCITY{qq{$I_VAL[_I_LOCATION]}}}[_T_TCITY_FLD_1];
    $O_VAL[_O_CITY_NAME] = $I_VAL[_I_CITY_NAME];
    $I_VAL[_I_STATE_NAME] = $$_TABLE_TSTATE{qq{@{[ ${$$_TABLE_TCITY{qq{$I_VAL[_I_LOCATION]}}}[_T_TCITY_FLD_2]
]}}} || $$_TABLE_TSTATE{qq{$I_VAL[_I_LOCATION]}};
```

```
    $O_VAL[_O_STATE_NAME] = $I_VAL[_I_STATE_NAME];
    $I_VAL[_I_STATE_NAME_UC] = uc($$_TABLE_TSTATE{qq{@{[ uc(${$$_TABLE_TCITY{qq{@{[ uc($I_VAL[_I_LOCATION]) ]}
}}}[_T_TCITY_FLD_2]) ]}}} || $$_TABLE_TSTATE{qq{$I_VAL[_I_LOCATION]}});
    $O_VAL[_O_STATE_NAME_UC] = $I_VAL[_I_STATE_NAME_UC];
    $I_VAL[_I_STATE_CODE] = ${$$_TABLE_TCITY{qq{@{[ uc($I_VAL[_I_LOCATION]) ]}}}}[_T_TCITY_FLD_2] || $I_VAL[_I
_LOCATION];
    $O_VAL[_O_STATE_CODE] = $I_VAL[_I_STATE_CODE];
    $I_VAL[_I_STATE_NAME_3] = $$_TABLE_TSTATE{qq{$I_VAL[_I_STATE_CODE]}};
    $O_VAL[_O_STATE_NAME_3] = $I_VAL[_I_STATE_NAME_3];
    print
        $O_VAL[_O_LOCATION],
        $O_VAL[_O_CITY_NAME],
        $O_VAL[_O_STATE_NAME],
        $O_VAL[_O_STATE_NAME_UC],
        $O_VAL[_O_STATE_CODE],
        $O_VAL[_O_STATE_NAME_3]
    ;
}

print STDERR ’[simple_tables.pql ’ . localtime() . "] $. records.";
my $benchmark_end = new Benchmark;
my $benchmark_timediff = timediff($benchmark_start, $benchmark_end);
print STDERR ’[simple_tables.pql ’ . localtime() . "] Code statistics: @{[timestr($benchmark_timediff)]}";
#-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
#++++++ Table TCITY --> Type :Pequel::Type::Table::Local ++++++
sub InitLookupTCITY
{
    my %_TABLE_TCITY;
    %_TABLE_TCITY =
    (
        ’ALIC’ => [’Alice Springs’, ’NT’],
        ’MEL’ => [’Melbourne’, ’VIC’],
        ’PER’ => [’Perth’, ’WA’],
        ’SYD’ => [’Sydney’, ’NSW’]
    );
    return \%_TABLE_TCITY;
}

#++++++ Table TSTATE --> Type :Pequel::Type::Table::Local ++++++
sub InitLookupTSTATE
{
    my %_TABLE_TSTATE;
    %_TABLE_TSTATE =
    (
        ’NSW’ => ’New South Wales’,
        ’NT’ => ’Northern Territory’,
        ’QLD’ => ’Queensland’,
        ’SA’ => ’South Australia’,
        ’VIC’ => ’Victoria’,
        ’WA’ => ’Western Australia’
    );
    return \%_TABLE_TSTATE;
}

sub PrintHeader
{
    local $\="\n";
    local $,="|";
    print
        ’LOCATION’,
        ’CITY_NAME’,
        ’STATE_NAME’,
        ’STATE_NAME_UC’,
        ’STATE_CODE’,
        ’STATE_NAME_3’
        ;
}
```

7. ABOUT PEQUEL

This document was generated by Pequel.

https://sourceforge.net/projects/pequel/

COPYRIGHT