



# examples/pequel\_tables.pql

by *Pequel*

---

[sample@youraddress.com](mailto:sample@youraddress.com)

## Pequel Tables Example Script

2.3



# Table of Contents

## Pequel Tables Example Script

SCRIPT NAME	1
DESCRIPTION	1
1. PROCESS DETAILS	1
1.1 PRODUCT_CODE	1
Description	1
1.2 PRODUCT_SALES_TOTAL	1
Description	1
Derived Input Field Evaluation	1
1.3 LOCATION	1
Description	1
1.4 LOCATION_SALES_TOTAL	1
Description	1
Derived Input Field Evaluation	1
1.5 COMMENT	1
Description	1
Derived Input Field Evaluation	2
2. CONFIGURATION SETTINGS	3
2.1 prefix	3
2.2 pequeldoc	3
2.3 detail	3
2.4 script_name	3
2.5 header	3
2.6 optimize	3
2.7 doc_title	3
2.8 doc_email	3
2.9 doc_version	3
3. TABLES	4
3.1 TSALESBYLOC	4
3.2 TSALESBYPROD	4
4. TABLE INFORMATION SUMMARY	5
4.1 Table List Sorted By Table Name	5
5. EXAMPLES/PEQUEL_TABLES.PQL	6
options	6
description	6
load table	6
input section	6
output section	6
6. PEQUEL GENERATED PROGRAM	7
7. ABOUT PEQUEL	14
COPYRIGHT	14



## SCRIPT NAME

examples/pequel\_tables.pql

## DESCRIPTION

This script demonstrates the use of pequel tables. This script contains a 'load table pequel' section. The tables specified in this section will have their data loaded by executing the pequel script specified. The field names for the table columns are as per the script output format. The output format for a script can be displayed with the '-list output\_format' option on the command line. It is important that any Pequel script used in the 'load table pequel' to load a table must have an input\_file option specification.

## 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 **5** fields. Fields are delimited by the '|' character.

### 1.1 *PRODUCT\_CODE*

Output Field

#### *Description*

Set to input field **PRODUCT\_CODE**

### 1.2 *PRODUCT\_SALES\_TOTAL*

Output Field

#### *Description*

Set to input field **SALESBYPROD**

#### *Derived Input Field Evaluation*

```
=> %TSALESBYPROD( PRODUCT_CODE )->SALES_TOTAL
```

### 1.3 *LOCATION*

Output Field

#### *Description*

Set to input field **LOCATION**

### 1.4 *LOCATION\_SALES\_TOTAL*

Output Field

#### *Description*

Set to input field **SALESBYLOC**

#### *Derived Input Field Evaluation*

```
=> %TSALESBYLOC( LOCATION )->SALES_TOTAL
```

### 1.5 *COMMENT*

Output Field

#### *Description*

Set to input field **COMMENT**

***Derived Input Field Evaluation***

```
=> %TSALESBYLOC(LOCATION)->TOP_PRODUCT eq PRODUCT_CODE  
?   '**Best Seller'  
:   ''
```

## 2. CONFIGURATION SETTINGS

### 2.1 *prefix*

directory pathname prefix.: examples

### 2.2 *pequeldoc*

generate pod / pdf pequel script Reference Guide.: pdf

### 2.3 *detail*

Include Pequel Generated Program chapter in Pequeldoc: 1

### 2.4 *script\_name*

script filename: examples/pequel\_tables.pql

### 2.5 *header*

write header record to output.: 1

### 2.6 *optimize*

optimize generated code.: 1

### 2.7 *doc\_title*

document title.: Pequel Tables Example Script

### 2.8 *doc\_email*

document email entry.: sample@youraddress.com

### 2.9 *doc\_version*

document version for pequel script.: 2.3

### 3. TABLES

#### 3.1 *TSALESBYLOC*

Table Type: ***external***

Data Source Filename: ***examples/sales\_ttl\_by\_loc.pql***

Key Field Number: ***1***

**3.1.1 *SALES\_TOTAL* = 2**

**3.1.2 *TOP\_PRODUCT* = 3**

#### 3.2 *TSALESBYPROD*

Table Type: ***external***

Data Source Filename: ***examples/sales\_ttl\_by\_prod.pql***

Key Field Number: ***1***

**3.2.1 *SALES\_TOTAL* = 2**



## 4. TABLE INFORMATION SUMMARY

### 4.1 Table List Sorted By Table Name

TSALESBYLOC — **1** (*external*)

TSALESBYPROD — **2** (*external*)

## 5. EXAMPLES/PEQUEL\_TABLES.PQL

### options

```
prefix(examples)
pequeldoc(pdf)
detail(1)
script_name(examples/pequel_tables.pql)
header(1)
optimize(1)
doc_title(Pequel Tables Example Script)
doc_email(sample@youraddress.com)
doc_version(2.3)
```

### description

This script demonstrates the use of pequel tables. This script contains a 'load table pequel' section. The tables specified in this section will have their data loaded by executing the pequel script specified. The field names for the table columns are as per the script output format. The output format for a script can be displayed with the '-list output\_format' option on the command line. It is important that any Pequel script used in the 'load table pequel' to load a table must have an input\_file option specification.

### load table

```
TSALESBYLOC /* Table Name */ \
  examples/sales_ttl_by_loc.pql /* Data Source Filename */ \
  1 /* Key Column Number */ \
  \
  SALES_TOTAL = 2 \
  TOP_PRODUCT = 3

TSALESBYPROD /* Table Name */ \
  examples/sales_ttl_by_prod.pql /* Data Source Filename */ \
  1 /* Key Column Number */ \
  \
  SALES_TOTAL = 2
```

### input section

```
PRODUCT_CODE
COST_PRICE
DESCRIPTION
SALES_CODE
SALES_PRICE
SALES_QTY
SALES_DATE
LOCATION
SALESBYLOC => %TSALESBYLOC(LOCATION)->SALES_TOTAL

SALESBYPROD => %TSALESBYPROD(PRODUCT_CODE)->SALES_TOTAL

COMMENT => %TSALESBYLOC(LOCATION)->TOP_PRODUCT eq PRODUCT_CODE
?   '**Best Seller'
:   ''
```

### output section

string	PRODUCT_CODE	PRODUCT_CODE
decimal	PRODUCT_SALES_TOTAL	SALESBYPROD
string	LOCATION	LOCATION
decimal	LOCATION_SALES_TOTAL	SALESBYLOC
string	COMMENT	COMMENT

## 6. PEQUEL GENERATED PROGRAM

```
#!/usr/bin/perl
#-----
# vim: syntax=perl ts=4 sw=4
#-----
#Generated By: pequel Version 2.4-5, Build: Wednesday November 16 21:56:42 GMT 2005
#           : http://sourceforge.net/projects/pequel/
#Script Name : pequel_tables.pql
#Created On  : Wed Nov 16 14:18:02 2005
#Perl Version: /usr/bin/perl 5.6.1 on solaris
#For         :
#-----
#Options:
#prefix(examples) directory pathname prefix.
#pequeldoc(pdf) generate pod / pdf pequel script Reference Guide.
#detail(1) Include Pequel Generated Program chapter in Pequeldoc
#script_name(examples/pequel_tables.pql) script filename
#header(1) write header record to output.
#optimize(1) optimize generated code.
#doc_title(Pequel Tables Example Script) document title.
#doc_email(sample@youraddress.com) document email entry.
#doc_version(2.3) document version for pequel script.
#-----
use strict;
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_SALESBYLOC        => int    8;
use constant _I_SALESBYPROD       => int    9;
use constant _I_COMMENT           => int   10;
use constant _O_PRODUCT_CODE      => int    1;
use constant _O_PRODUCT_SALES_TOTAL => int    2;
use constant _O_LOCATION          => int    3;
use constant _O_LOCATION_SALES_TOTAL => int    4;
use constant _O_COMMENT           => int    5;
use constant _T_TSALSBYLOC_FLD_SALES_TOTAL => int    0;
use constant _T_TSALSBYLOC_FLD_TOP_PRODUCT => int    1;
use constant _T_TSALSBYPROD_FLD_SALES_TOTAL => int    0;
use constant _I_TSALSBYLOC_LOCATION_FLD_KEY => int   11;
use constant _I_TSALSBYLOC_LOCATION_FLD_SALES_TOTAL => int   12;
use constant _I_TSALSBYLOC_LOCATION_FLD_TOP_PRODUCT => int   13;
use constant _I_TSALSBYPROD_PRODUCT_CODE_FLD_KEY => int   14;
use constant _I_TSALSBYPROD_PRODUCT_CODE_FLD_SALES_TOTAL => int   15;
local $\="\\n";
local $,="|";
print STDERR "[examples/pequel_tables.pql ' . localtime() . "] Init";
use constant VERBOSE => int 10000;
use constant LAST_ICELL => int 10;
my @I_VAL;
my @O_VAL;
my $_inprec=0;
foreach my $f (1..5) { $_O_VAL[$f] = undef; }
my $_TABLE_TSALSBYLOC = &LoadTableTSALSBYLOC; # ref to %TSALSBYLOC hash
my $_TABLE_TSALSBYPROD = &LoadTableTSALSBYPROD; # ref to %TSALSBYPROD hash
&PrintHeader();
print STDERR "[examples/pequel_tables.pql ' . localtime() . "] Start";
use Benchmark;
my $benchmark_start = new Benchmark;
while (<STDIN>)
{
    ++$_inprec;
    print STDERR "[examples/pequel_tables.pql ' . localtime() . "] $_inprec records." if ($_inprec % VERBOSE
    == 0);
    chomp;
    @I_VAL = split("[|]", $_);
    $_O_VAL[_O_PRODUCT_CODE] = $_I_VAL[_I_PRODUCT_CODE];
    $_I_VAL[_I_SALESBYPROD] = $_$_TABLE_TSALSBYPROD{qq{$_I_VAL[_I_PRODUCT_CODE]}};
    $_O_VAL[_O_PRODUCT_SALES_TOTAL] = $_I_VAL[_I_SALESBYPROD];
    $_O_VAL[_O_LOCATION] = $_I_VAL[_I_LOCATION];
    $_I_VAL[_I_SALESBYLOC] = $_$_TABLE_TSALSBYLOC{qq{$_I_VAL[_I_LOCATION]}}[_T_TSALSBYLOC_FLD_SALES_TOTAL];
    $_O_VAL[_O_LOCATION_SALES_TOTAL] = $_I_VAL[_I_SALESBYLOC];
    $_I_VAL[_I_COMMENT] = $_$_TABLE_TSALSBYLOC{qq{$_I_VAL[_I_LOCATION]}}[_T_TSALSBYLOC_FLD_TOP_PRODUCT] eq $_
    I_VAL[_I_PRODUCT_CODE] ? '**Best Seller' : '';
    $_O_VAL[_O_COMMENT] = $_I_VAL[_I_COMMENT];
    print STDOUT
        $_O_VAL[_O_PRODUCT_CODE],
```

```

        $O_VAL[_O_PRODUCT_SALES_TOTAL],
        $O_VAL[_O_LOCATION],
        $O_VAL[_O_LOCATION_SALES_TOTAL],
        $O_VAL[_O_COMMENT]
    }
}

close(STDIN);
print STDERR '[examples/pequel_tables.pql ' . localtime() . "] $_inprec records.";
my $benchmark_end = new Benchmark;
my $benchmark_timediff = timediff($benchmark_start, $benchmark_end);
print STDERR '[examples/pequel_tables.pql ' . localtime() . "] Code statistics: @ {[timestr($benchmark_timediff
)}}";
#-----
#+++++ Table TSALESBYLOC --> Type :ETL::Pequel::Type::Table::External::Pequel +++++
sub LoadTableTSALESBYLOC
{
    my $_TABLE_TSALESBYLOC;
    print STDERR '[examples/pequel_tables.pql ' . localtime() . "] Loading table TSALESBYLOC from examples/sal
es_ttl_by_loc.pql...";
    my $pid = open(TSALESBYLOC, '-|'); # Fork
    my $count=0;
    if ($pid) # Parent
    {
        while (<TSALESBYLOC>)
        {
            chomp;
            my (@flds) = split("[|]", $_, -1);
            $_TABLE_TSALESBYLOC{$flds[0]} = [ @flds[ 1,2 ]];
            print STDERR '[examples/pequel_tables.pql ' . localtime() . "] Table TSALESBYLOC $. records..." if
($_. % 100000 == 0);
        }

        $count=$_;
        close(TSALESBYLOC);
    }

    else # Child
    {
        &p_LoadTableTSALESBYLOC::LoadTableTSALESBYLOC;
        exit(0);
    }

    print STDERR '[examples/pequel_tables.pql ' . localtime() . "] Table TSALESBYLOC loaded $count records.";
    close(TSALESBYLOC);
    return \$_TABLE_TSALESBYLOC;
}

{
    package p_LoadTableTSALESBYLOC;
    sub LoadTableTSALESBYLOC
    {
        #!/usr/bin/perl
#-----
#       vim: syntax=perl ts=4 sw=4
#-----
#       Generated By: pequel Version 2.4-5, Build: Wednesday November 16 21:56:42 GMT 2005
#       : http://sourceforge.net/projects/pequel/
#       Script Name : sales_ttl_by_loc.pql
#       Created On  : Wed Nov 16 14:17:58 2005
#       Perl Version: /usr/bin/perl 5.6.1 on solaris
#       For       :
#-----
#       Options:
#       input_file(sample.data) input data filename
#       header(1) write header record to output.
#       optimize(1) optimize generated code.
#       hash(1) Generate in memory. Input data can be unsorted.
#       doc_title(Pequel Table Example Script) document title.
#       doc_email(sample@youraddress.com) document email entry.
#       doc_version(2.3) document version for pequel script.
#-----
        use strict;
        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_SALES_TOTAL     => int    8;
        use constant _I_TOP_PRODUCT     => int    9;
        use constant _O_LOCATION        => int    1;
        use constant _O_SALES_TOTAL     => int    2;
    }
}

```

```

use constant _O_TOP_PRODUCT      => int    3;
use constant _T_TTOPPRODBYLOC_FLD_PRODUCT_CODE  => int    0;
use constant _I_TTOPPRODBYLOC_LOCATION_FLD_KEY   => int   10;
use constant _I_TTOPPRODBYLOC_LOCATION_FLD_PRODUCT_CODE  => int   11;
local $\'=\'\n";
local $,="|";
print STDERR '[examples/sales_ttl_by_loc.pql ' . localtime() . "] Init";
use constant VERBOSE => int 10000;
use constant LAST_ICELL => int 9;
my @I_VAL;
my %O_VAL;
my $key;
my $_inprec=0;
my $_TABLE_TTOPPRODBYLOC = &LoadTableTTOPPRODBYLOC; # ref to %$_TABLE_TTOPPRODBYLOC hash
open(DATA, q{examples/sample.data}) || die "Cannot open examples/sample.data: $!";
&PrintHeader();
print STDERR '[examples/sales_ttl_by_loc.pql ' . localtime() . "] Start";
use Benchmark;
my $benchmark_start = new Benchmark;
while (<DATA>)
{
    ++$_inprec;
    print STDERR '[examples/sales_ttl_by_loc.pql ' . localtime() . "] $_inprec records." if ($_inprec
s % VERBOSE
    == 0);
    chomp;
    @I_VAL = split("[|]", $_);
    $key = ( $_I_VAL[_I_LOCATION] );
    $O_VAL{$key}{_O_LOCATION} = $_I_VAL[_I_LOCATION];
    $_I_VAL[_I_SALES_TOTAL] = $_I_VAL[_I_SALES_QTY] * $_I_VAL[_I_SALES_PRICE];
    $O_VAL{$key}{_O_SALES_TOTAL} += $_I_VAL[_I_SALES_TOTAL] unless ($_I_VAL[_I_SALES_TOTAL] eq '');
    $_I_VAL[_I_TOP_PRODUCT] = $_TABLE_TTOPPRODBYLOC{qq{$_I_VAL[_I_LOCATION]}};
    $O_VAL{$key}{_O_TOP_PRODUCT} = $_I_VAL[_I_TOP_PRODUCT];
}

foreach $key (sort keys %O_VAL)
{
    print STDOUT
        $O_VAL{$key}{_O_LOCATION},
        $O_VAL{$key}{_O_SALES_TOTAL},
        $O_VAL{$key}{_O_TOP_PRODUCT}
    ;
}

close(DATA);
print STDERR '[examples/sales_ttl_by_loc.pql ' . localtime() . "] $_inprec records.";
my $benchmark_end = new Benchmark;
my $benchmark_timediff = timediff($benchmark_start, $benchmark_end);
print STDERR '[examples/sales_ttl_by_loc.pql ' . localtime() . "] Code statistics: @{{timestr($benchma
rk_timediff)}}";
#-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
#   ++++++ Table TTOPPRODBYLOC --> Type :ETL::Pequel::Type::Table::External::Pequel ++++++
sub LoadTableTTOPPRODBYLOC
{
    my $_TABLE_TTOPPRODBYLOC;
    print STDERR '[examples/sales_ttl_by_loc.pql ' . localtime() . "] Loading table TTOPPRODBYLOC from
examples/top_prod_by_loc.pql...";
    my $pid = open(TTOPPRODBYLOC, '-|'); # Fork
    my $count=0;
    if ($pid) # Parent
    {
        while (<TTOPPRODBYLOC>)
        {
            chomp;
            my (@flds) = split("[|]", $_, -1);
            $_TABLE_TTOPPRODBYLOC{$_flds[0]} = $_flds[ 1 ];
            print STDERR '[examples/sales_ttl_by_loc.pql ' . localtime() . "] Table TTOPPRODBYLOC $. r
ecords..." if ($. % 100000 == 0);
        }

        $count=$.;
        close(TTOPPRODBYLOC);
    }

    else # Child
    {
        &p_LoadTableTTOPPRODBYLOC::LoadTableTTOPPRODBYLOC;
        exit(0);
    }

    print STDERR '[examples/sales_ttl_by_loc.pql ' . localtime() . "] Table TTOPPRODBYLOC loaded $coun
t records.";
    close(TTOPPRODBYLOC);
    return \$_TABLE_TTOPPRODBYLOC;
}

```

```

{
    package p_LoadTableTTOPPRODBYLOC;
    sub LoadTableTTOPPRODBYLOC
    {
#
# !/usr/bin/perl
#-----
#      vim: syntax=perl ts=4 sw=4
#-----
#      Generated By: pequel Version 2.4-5, Build: Wednesday November 16 21:56:42 GMT 2005
#      : http://sourceforge.net/projects/pequel/
#      Script Name : top_prod_by_loc.pql
#      Created On  : Wed Nov 16 14:17:57 2005
#      Perl Version: /usr/bin/perl 5.6.1 on solaris
#      For        :
#-----
#      Options:
#          input_file(sample.data) input data filename
#          header(1) write header record to output.
#          optimize(1) optimize generated code.
#          hash(1) Generate in memory. Input data can be unsorted.
#          doc_title(Pequel Table Example Script) document title.
#          doc_email(sample@youraddress.com) document email entry.
#          doc_version(2.3) document version for pequel script.
#-----
        use strict;
        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_SALES_TOTAL     => int    8;
        use constant _O_LOCATION        => int    1;
        use constant _O_MAXSALES        => int    2;
        use constant _O_PRODUCT_CODE    => int    3;
        local $="\n";
        local $,="|";
        print STDERR "[examples/top_prod_by_loc.pql ' . localtime() . "] Init";
        use constant VERBOSE => int 10000;
        use constant LAST_ICELL => int 8;
        my @I_VAL;
        my %O_VAL;
        my $key;
        my $_inprec=0;
        open(DATA, q{examples/sample.data}) || die "Cannot open examples/sample.data: $!";
        &PrintHeader();
        print STDERR "[examples/top_prod_by_loc.pql ' . localtime() . "] Start";
        use Benchmark;
        my $benchmark_start = new Benchmark;
        while (<DATA>)
        {
            ++$_inprec;
            print STDERR "[examples/top_prod_by_loc.pql ' . localtime() . "] $_inprec records." if ($
_inprec % VERBOSE == 0);
            chomp;
            @I_VAL = split("[|]", $_);
            $key = ( $I_VAL[_I_LOCATION] );
            $O_VAL{$key}{_O_LOCATION} = $I_VAL[_I_LOCATION];
            $I_VAL[_I_SALES_TOTAL] = $I_VAL[_I_SALES_QTY] * $I_VAL[_I_SALES_PRICE];
            $O_VAL{$key}{_O_MAXSALES} = $I_VAL[_I_SALES_TOTAL]
                if (!defined($O_VAL{$key}{_O_MAXSALES}) || $I_VAL[_I_SALES_TOTAL] > $O_VAL{$key}{_O_
MAXSALES});

            if (sprintf("%.2f", $I_VAL[_I_SALES_TOTAL]) eq sprintf("%.2f", $O_VAL{$key}{_O_MAXSALES}))
        {
            $O_VAL{$key}{_O_PRODUCT_CODE} = $I_VAL[_I_PRODUCT_CODE] if (!defined($O_VAL{$key}{_O_P
RODUCT_CODE}));
        }
        }

        foreach $key (sort keys %O_VAL)
        {
            print STDOUT
                $O_VAL{$key}{_O_LOCATION},
                $O_VAL{$key}{_O_PRODUCT_CODE}
            ;
        }

        close(DATA);
        print STDERR "[examples/top_prod_by_loc.pql ' . localtime() . "] $_inprec records.";
        my $benchmark_end = new Benchmark;
        my $benchmark_timediff = timediff($benchmark_start, $benchmark_end);
        print STDERR "[examples/top_prod_by_loc.pql ' . localtime() . "] Code statistics: @[{timestr($

```

```

benchmark_timediff)}}";
#+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
        sub PrintHeader
        {
            local $\\="\n";
            local $,="|";
            print STDOUT
                'LOCATION',
                'PRODUCT_CODE'
                ;
        }

    }

    sub PrintHeader
    {
        local $\\="\n";
        local $,="|";
        print STDOUT
            'LOCATION',
            'SALES_TOTAL',
            'TOP_PRODUCT'
            ;
    }

}

}

#+----- Table TSALESBYPROD --> Type :ETL::Pequel::Type::Table::External::Pequel +-----
sub LoadTableTSALESBYPROD
{
    my $_TABLE_TSALESBYPROD;
    print STDERR '[examples/pequel_tables.pql ' . localtime() . "] Loading table TSALESBYPROD from examples/sa
les_ttl_by_prod.pql...";
    my $pid = open(TSALESBYPROD, '-|'); # Fork
    my $count=0;
    if ($pid) # Parent
    {
        while (<TSALESBYPROD>)
        {
            chomp;
            my (@flds) = split("[|]", $_, -1);
            $_TABLE_TSALESBYPROD{$flds[0]} = $flds[ 1 ];
            print STDERR '[examples/pequel_tables.pql ' . localtime() . "] Table TSALESBYPROD $. records..." i
f ($. % 100000 == 0);
        }

        $count=$_.
        close(TSALESBYPROD);
    }

    else # Child
    {
        &p_LoadTableTSALESBYPROD::LoadTableTSALESBYPROD;
        exit(0);
    }

    print STDERR '[examples/pequel_tables.pql ' . localtime() . "] Table TSALESBYPROD loaded $count records.";
    close(TSALESBYPROD);
    return \$_TABLE_TSALESBYPROD;
}

{
    package p_LoadTableTSALESBYPROD;
    sub LoadTableTSALESBYPROD
    {
        # !usr/bin/perl
        # vim: syntax=perl ts=4 sw=4
        # Generated By: pequel Version 2.4-5, Build: Wednesday November 16 21:56:42 GMT 2005
        #           : http://sourceforge.net/projects/pequel/
        # Script Name : sales_ttl_by_prod.pql
        # Created On  : Wed Nov 16 14:18:00 2005
        # Perl Version: /usr/bin/perl 5.6.1 on solaris
        # For       :
        # Options:
        #     input_file(sample.data) input data filename
        #     header(1) write header record to output.
        #     optimize(1) optimize generated code.
        #     doc title(Pequel Table Example Script) document title.
    }
}

```

```

# doc_email(sample@youraddress.com) document email entry.
# doc_version(2.3) document version for pequel script.
#-----
use strict;
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_SALES_TOTAL => int 8;
use constant _O_PRODUCT_CODE => int 1;
use constant _O_SALES_TOTAL => int 2;
local $\\="\n";
local $,="|";
print STDERR '[examples/sales_ttl_by_prod.pql ' . localtime() . "] Init";
use constant VERBOSE => int 10000;
use constant LAST_ICELL => int 8;
my @I_VAL;
my @O_VAL;
my $_inprec=0;
my $key__I_PRODUCT_CODE;
my $previous_key__I_PRODUCT_CODE = undef;
foreach my $f (1..2) { $O_VAL[$f] = undef; }
open(DATA, q{examples/sample.data}) || die "Cannot open examples/sample.data: $!";
&PrintHeader();
print STDERR '[examples/sales_ttl_by_prod.pql ' . localtime() . "] Start";
use Benchmark;
my $benchmark_start = new Benchmark;
while (<DATA>)
{
    ++$_inprec;
    print STDERR '[examples/sales_ttl_by_prod.pql ' . localtime() . "] $_inprec records." if ($_inpre
cs % VERBOSE == 0);
    chomp;
    @I_VAL = split("[|]", $_);
    $key__I_PRODUCT_CODE = $I_VAL[_I_PRODUCT_CODE];
    if (!defined($previous_key__I_PRODUCT_CODE))
    {
        $previous_key__I_PRODUCT_CODE = $key__I_PRODUCT_CODE;
    }

    elsif ($previous_key__I_PRODUCT_CODE ne $key__I_PRODUCT_CODE)
    {
        print STDOUT
            $O_VAL[_O_PRODUCT_CODE],
            $O_VAL[_O_SALES_TOTAL]
        ;
        $previous_key__I_PRODUCT_CODE = $key__I_PRODUCT_CODE;
        @O_VAL = undef;
    }

    $O_VAL[_O_PRODUCT_CODE] = $I_VAL[_I_PRODUCT_CODE];
    $I_VAL[_I_SALES_TOTAL] = $I_VAL[_I_SALES_QTY] * $I_VAL[_I_SALES_PRICE];
    $O_VAL[_O_SALES_TOTAL] += $I_VAL[_I_SALES_TOTAL] unless ($I_VAL[_I_SALES_TOTAL] eq '');
}

print STDOUT
    $O_VAL[_O_PRODUCT_CODE],
    $O_VAL[_O_SALES_TOTAL]
;
close(DATA);
print STDERR '[examples/sales_ttl_by_prod.pql ' . localtime() . "] $_inprec records.";
my $benchmark_end = new Benchmark;
my $benchmark_timediff = timediff($benchmark_start, $benchmark_end);
print STDERR '[examples/sales_ttl_by_prod.pql ' . localtime() . "] Code statistics: @{"timestr($benchm
ark_timediff)}";
#-----
sub PrintHeader
{
    local $\\="\n";
    local $,="|";
    print STDOUT
        'PRODUCT_CODE',
        'SALES_TOTAL'
    ;
}

}

sub PrintHeader

```



```
{
  local $\="\n";
  local $,="| ";
  print STDOUT
    'PRODUCT_CODE' ,
    'PRODUCT_SALES_TOTAL' ,
    'LOCATION' ,
    'LOCATION_SALES_TOTAL' ,
    'COMMENT'
    ;
}
```

## 7. ABOUT PEQUEL

This document was generated by Pequel.

*<https://sourceforge.net/projects/pequel/>*

## COPYRIGHT

Copyright ©1999-2005, Mario Gaffiero. All Rights Reserved.

'Pequel' TM Copyright ©1999-2005, Mario Gaffiero. All Rights Reserved.

This program and all its component contents is copyrighted free software by Mario Gaffiero and is released under the GNU General Public License (GPL), Version 2, a copy of which may be found at <http://www.opensource.org/licenses/gpl-license.html>

Pequel is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

Pequel is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with Pequel; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA



