Objective:

- design a warehouse database using star schema design
- load a data warehouse table using SQL against the production table

Background

Heather Sweeney Dodge is a home redesign company that specializes in home remodel. They give seminars around the country and sell books and video at the seminars. They track attendance and sales at the seminars in a database. They would like to analyze sales trends and find out which of their products is most popular and which of their important customers.

Part 1: Create the HSD tables and load data

Download files

- **`HSD CreateTablesWithData.sql`**, the production tables for the HSD (Heather Sweeney Dodge Company) and
- **`HSDDW CreateTablesWithData.sql,`** the data warehouse tables.

Part 2: Create the HSDDW (HSD Data Warehouse) tables

Execute the script HSD CreateTablesWithData script to create the production table and data.

Execute the script HSDDW CreateTablesWithData which create the warehouse tables and will loads the HSDDW tables with data from the HSD tables. Study the script carefully to understand how the data from HSD tables is copied and transformed and loaded into the HSDDW tables. This is known as ETL (Extract – Transform -Load). ETL may be done by sql scripts in simple cases (such as here). In more complex cases it needs special application programs to filter and "scrub" the data.

Part 3: Modifying the HSD_DW warehouse tables

1. Examine the sql statements in the **`HSDDW CreateTablesWithData.sql`** What transformations of data were made when HSD-DW was loaded with data? [Identify all the transformations, showing the tables and columns of the HSD data and how they are mapped into the HSD-DW database.

Copy your answer into hw5.sql file

Create the SALEs_FOR_RFM table to the HSD-DW database using the create table statement below. RFM refers to "Recency, Frequency, Monetary Value" which are ways to identify important customers.

```
create table sales_for_rfm (
     TimeId              int  not null,
     CustomerId          int  not null,
     InvoiceNumber  int  not null,
     PreTaxTotalSales Numeric(9,2) not null,
       constraint sales_for_rfm_pk
       primary key(TimeId, CustomerId, InvoiceNumber),
     constraint srfm_timeline_fk foreign key (TimeId)
       references timeline(TimeId)
        on update no action
        on delete no action,
     constraint srfm_customer_fk foreign key (CustomerId)
       references customer(CustomerId)
        on update no action
        on delete no action
);
```

2. What data will be used to load the SALES_FOR_RFM fact able? Write the complete set of SQL statements necessary to load the data. Then run the insert statement to populate the SALES_FOR_RFM table. Your INSERT statement will be similar to the load product sales INSERT statement on line 124 of HSDDW CreateTablesWithData.sql

```
Copy your insert statement into hw5.sql file
```

A query to summarize product units sold by Customer (CustomerName) , City, and Product(ProductName) and Year would be :

```
SELECT c.CustomerId, c.CustomerName, c.City,
       p.ProductNumber, p.ProductName,
       t.Year,t.QuarterText,
       SUM(ps.Quantity) AS TotalQuantity
 FROM customer c, product_sales ps, product p,
      timeline t
WHERE c.CustomerId = ps.CustomerID
      AND p.ProductNumber = ps.ProductNumber
      AND t.TimeId = ps.TimeID
GROUP BY c.CustomerId, c.CustomerName, c.City,
         p.ProductNumber, p.ProductName,
         t.QuarterText, t.Year
ORDER BY c.CustomerName, t.Year, t.QuarterText;
```

In the above query, product_sales table is called the ***fact table***; customer, timeline and product tables are called ***dimension tables***.

3. Modify the above query to report the total dollar amount **of each product for each year** instead of the quantity sold for each product. Create an SQL View equivalent of the SQL query you wrote. Use your view and verify that the view returns the same number of rows in the same order as your sql SELECT.

Copy your create view statement to hw5.sql

Define a PAYMENT_TYPE dimension table as

```
create table hsddw.payment_type (
   payment_type_id     int  not null primary key,
   payment_type varchar(35) not null
);
```

and insert the following values

```
insert into payment_type values
(1, 'VISA'),
(2, 'MasterCard'),
(3, 'Cash'),
(4, 'American Express'),
(5, 'PayPal'),
(6, 'Check');
```

4. Modify the file HSDDW CreateTablesWithData.sql
   - **a.** add a column to the create product_sales on line 34 for **payment_id.**
   - b. modify the INSERT statement on line 124 to populate the product_sales table including the new payment_id column.
   - c. rerun the HSDDW CreateTablesWithData.sql script and verify that payment_id has been populated. You should have 48 rows in the product_sales table.

Copy your modified insert statement to hw5.sql

What to submit for this assignment?

The modified hw5.sql file.