Back to Resume

Creating Views to manipulate the data warehouse

Notes: This was a copy of a recreated OLAP Data Warehouse with several kinds of purposeful flaws that I was given. I created it again using the code below that I made. When solving the various problems and challenges that the data warehouse has, I decided that the best practice was to use Views to manipulate the data without changing any of the database structure.

```
/*************** Tabular Modeling ******************/
/*********** Create DWSampleData Database **********/      All Code was written by: Christopher Singleton
/***************************************************/   (Data Warehouse "DWSampleData" was recreated)
USE master;
GO
IF EXISTS (SELECT name FROM sys.databases WHERE name = N'DWSampleData')
 BEGIN
 ALTER DATABASE [DWSampleData] SET SINGLE_USER WITH ROLLBACK IMMEDIATE
--====================== Drop This Database and End Function ======================--
 DROP DATABASE [DWSampleData] --If it already exists so we can start fresh.
/*Print out that the database was dropped */
 /*Convert to sysdatetime and then cast to varchar. */
 PRINT 'DWSampleData Database: Dropped Database Successfully.'
 + CAST(CONVERT(varchar, SYSDATETIME(), 121) AS varchar (20))
 END
--================== Create the Database with Settings ======================--
--Its always a good idea to plan the size of the database for growth. --Plenty of size...
CREATE DATABASE [DWSampleData] ON PRIMARY
 (NAME = N'DWSampleData'
 , FILENAME = N'C:\_BISolutions\BUSIT 205\TheFinal\DWSampleData.mdf'
 --Store Database Here
 , SIZE = 10MB
 , MAXSIZE = 1GB
 , FILEGROWTH = 10MB)
 LOG ON
 (NAME = N'DWSampleData_log'
 --Store Log File Here
 , FILENAME = N'C:\_BISolutions\BUSIT 205\TheFinal\DWSampleData_log.LDF'
 , SIZE = 1MB
 , MAXSIZE = 1GB
 , FILEGROWTH = 10MB)
GO
/*Log In Owner Database Name = SA    Note: SA Means "System Administrator"*/
EXEC [DWSampleData].dbo.sp_changedbowner @loginame = N'SA', @map=false
```

```
GO

--===============Set's the Recovery Record Log Settings================--
 /*Note: Only use this mode "BULK_LOGGED" when there are no other users, otherwise data loss can happen. */
ALTER DATABASE [DWSampleData] SET RECOVERY BULK_LOGGED
 -- Below: Prints out the system Date Time with a message.
 --Print out that the database was created with Date/Time, CAST to varchar.
PRINT 'DWSampleData Database: Database was Created Successfully.'
 + CAST(CONVERT(varchar, SYSDATETIME(), 121) AS varchar (20))
 GO

 USE DWSampleData;
 GO

/*************** Tabular Modeling ****************/
/************ Create DWSampleData Tables ***********/
/***********************************************/

 --Drop the table first if it exists.
 IF OBJECT_ID('DimCategories', 'U') IS NOT NULL
 DROP TABLE [dbo].[DimCategories];
GO

CREATE TABLE DimCategories
 --PK-Set to row 1, increment 1.
 ([CategoryId] INT NOT NULL PRIMARY KEY IDENTITY(1,1)
 ,[CategoryName] VARCHAR(100) NOT NULL
 )
 GO

  --Drop the table first if it exists.
 IF OBJECT_ID('DimCustomers', 'U') IS NOT NULL
 DROP TABLE [dbo].[DimCustomers];
GO
```

```
CREATE TABLE DimCustomers
--PK-Set to row 1, increment 1.
([CustomerKey] INT NOT NULL PRIMARY KEY IDENTITY(1,1)
,[CustomerID] INT NOT NULL
,[CustomerName] NVARCHAR(100) NOT NULL
,[CustomerCity] NVARCHAR(100) NOT NULL
,[CustomerState] NVARCHAR(100) NOT NULL
)
GO

--SELECT * FROM DimCustomers

 --Drop the table first if it exists.
IF OBJECT_ID('DimDates', 'U') IS NOT NULL
DROP TABLE [dbo].[DimDates];
GO

CREATE TABLE DimDates
--PK-Set to row 1, increment 1.
([DateKey] INT NOT NULL PRIMARY KEY IDENTITY(20100101,1)
,[DateName] NVARCHAR(50) NOT NULL
,[MonthKey] INT NOT NULL
,[MonthName] NVARCHAR(50) NOT NULL
,[YearKey] INT NOT NULL
,[YearName] NVARCHAR(50) NOT NULL
)
GO

--Drop the table first if it exists.
IF OBJECT_ID('DimDepartments', 'U') IS NOT NULL
DROP TABLE [dbo].[DimDepartments];
GO
CREATE TABLE DimDepartments
            ([DepartmentKey] INT NOT NULL PRIMARY KEY IDENTITY(1,1)
            ,[DepartmentID] INT NOT NULL
            ,[DepartmentName] NVARCHAR(100) NOT NULL
            )
GO
```

```
IF OBJECT_ID('DimEmployees', 'U') IS NOT NULL
DROP TABLE [dbo].[DimEmployees];
GO
CREATE TABLE DimEmployees
               ([EmployeeKey] INT NOT NULL PRIMARY KEY IDENTITY(1,1)
                ,[EmployeeID] INT NOT NULL
                ,[EmployeeFullName] NVARCHAR(100) NOT NULL
                ,[DepartmentKey] INT NOT NULL --Foreign Key
                ,[ManagerKey] INT NOT NULL --Foreign Key
                )
GO

IF OBJECT_ID('DimProducts', 'U') IS NOT NULL
DROP TABLE [dbo].[DimProducts];
GO
CREATE TABLE DimProducts
               ([ProductKey] INT NOT NULL PRIMARY KEY IDENTITY(1,1)
               ,[ProductId] INT NOT NULL
               ,[ProductName] NVARCHAR(100) NOT NULL
               ,[ProductStandardPrice] DECIMAL(18,4) NULL
               )
GO

IF OBJECT_ID('FactInventories', 'U') IS NOT NULL
DROP TABLE [dbo].[FactInventories];
GO
CREATE TABLE FactInventories
               ([InventoryID] INT NOT NULL
               ,[ProductKey] INT NOT NULL
               ,[InventoryDateKey] INT NOT NULL
               ,[InventoryCounts] INT NOT NULL
               ,CONSTRAINT PK_FactInventories PRIMARY KEY(InventoryID
                                                          ,ProductKey
                                                          ,InventoryDateKey)
                )
GO
```

```
IF OBJECT_ID('FactProductCategories', 'U') IS NOT NULL
DROP TABLE [dbo].[FactProductCategories];
GO
CREATE TABLE FactProductCategories
            ([ProductKey] INT NOT NULL
            ,[CategoryKey] INT NOT NULL
            ,CONSTRAINT PK_FactProductCategories PRIMARY KEY(ProductKey
                                                            ,CategoryKey)
            )
GO

IF OBJECT_ID('FactSales', 'U') IS NOT NULL
DROP TABLE [dbo].[FactSales];
GO
CREATE TABLE FactSales
            ([SalesID] INT NOT NULL
            ,[SalesLineItemID] INT NOT NULL
            ,[EmployeeKey] INT NOT NULL
            ,[SalesDateKey] INT NOT NULL
            ,[ShipDateKey] INT NOT NULL
            ,[CustomerKey] INT NOT NULL
            ,[ProductKey] INT NOT NULL
            ,[SalesDollars] DECIMAL(18,4)
            ,[SalesUnits] INT NOT NULL
            ,CONSTRAINT PK_FactSales PRIMARY KEY(SalesID
                                                ,SalesLineItemID
                                                ,EmployeeKey
                                                ,SalesDateKey
                                                ,ShipDateKey
                                                ,CustomerKey
                                                ,ProductKey)
)
GO
/*Print out that the tables where successfully created */
/*Convert to sysdatetime and then cast to varchar. */
PRINT 'DWSampleData Table Creation: Created All Tables Successfully.'
+ CAST(CONVERT(varchar, SYSDATETIME(), 121) AS varchar (20))
GO
```

```
/*************** Tabular Modeling ****************/
/******** Create Relationship Foreign Keys *********/
/***************************************************/

ALTER TABLE DimEmployees
ADD CONSTRAINT FK_DimEmployees_DimDepartments
FOREIGN KEY (DepartmentKey)
REFERENCES DimDepartments (DepartmentKey)
GO
ALTER TABLE DimEmployees
ADD CONSTRAINT FK_DimEmployees_DimEmployees
FOREIGN KEY (ManagerKey)
REFERENCES DimEmployees (EmployeeKey)
GO
ALTER TABLE FactInventories
ADD CONSTRAINT FK_FactInventories_DimDates
FOREIGN KEY (InventoryDateKey)
REFERENCES DimDates (DateKey)
GO
ALTER TABLE FactInventories
ADD CONSTRAINT FK_FactInventories_DimProducts
FOREIGN KEY (ProductKey)
REFERENCES DimProducts (ProductKey)
GO
ALTER TABLE FactSales
ADD CONSTRAINT FK_FactSales_DimCustomers
FOREIGN KEY (CustomerKey)
REFERENCES DimCustomers (CustomerKey)
GO
ALTER TABLE FactSales
ADD CONSTRAINT FK_FactSales_DimDates
FOREIGN KEY (SalesDateKey)
REFERENCES DimDates (DateKey)
GO
ALTER TABLE FactSales
ADD CONSTRAINT FK_FactSales_DimDates1
FOREIGN KEY (ShipDateKey)
REFERENCES DimDates (DateKey)
```

```
GO
ALTER TABLE FactSales
ADD CONSTRAINT FK_FactSales_DimEmployees
FOREIGN KEY (EmployeeKey)
REFERENCES DimEmployees (EmployeeKey)
GO
ALTER TABLE FactSales
ADD CONSTRAINT FK_FactSales_DimProducts
FOREIGN KEY (ProductKey)
REFERENCES DimProducts (ProductKey)
GO

/*Print out that that the foreign keys were created successfully */
/*Convert to sysdatetime and then cast to varchar. */
PRINT 'DWSampleData Database: Created Foreign Keys Successfully.'
+ CAST(CONVERT(varchar, SYSDATETIME(), 121) AS varchar (20))
GO

/**************** Tabular Modeling *****************/
/************ Populate Tables with Data ************/
/*************************************************/
--================ DimCategories ================--

SET IDENTITY_INSERT DimCategories ON
GO

INSERT INTO DimCategories (CategoryId, CategoryName)
VALUES(100, 'CatA'),
          (200, 'CatB');

SELECT CategoryId
        ,CategoryName
FROM [DWSampleData].[dbo].[DimCategories]

SET IDENTITY_INSERT DimCategories OFF
GO
```

```
--================= DimCustomers =================--

INSERT INTO DimCustomers (CustomerID
                          ,CustomerName
                          ,CustomerCity
                          ,CustomerState)

VALUES(100, 'Dave Darsen', 'Seattle', 'Washington'),
      (200, 'Pat Peterson', 'Tacoma', 'Washington'),
      (300, 'Alex Allen', 'Portland', 'Oregon');

SELECT CustomerID
      ,CustomerName
      ,CustomerCity
      ,CustomerState
FROM [dbo].[DimCustomers]
GO

--=================== DimDates ===================--

/*Use AS needed:
 TRUNCATE TABLE DimDates
 SET IDENTITY_INSERT DimDates ON
*/
-- Create variables to hold the start and end date
DECLARE @StartDate datetime = '01/01/2010'
DECLARE @EndDate datetime = '12/31/2020'
-- Use a while loop to add dates to the table
DECLARE @DateInProcess datetime
SET @DateInProcess = @StartDate
WHILE @DateInProcess <= @EndDate
```

```
BEGIN
    --SET IDENTITY_INSERT DimDates ON
    -- Add a row into the date dimension table for this date
    INSERT INTO DimDates
    ( [DateName], [MonthKey], [MonthName], [YearKey], [YearName] )
    VALUES (DateName(weekday, @DateInProcess ) + ', ' + DateName(mm, @DateInProcess ) + ' ' + Convert(nVarchar(50)
                ,@DateInProcess, 110)
               ,Month( @DateInProcess ) -- [MonthID]
               ,DateName( month, @DateInProcess ) -- [MonthName]
               ,Year( @DateInProcess ) -- [YearID]
               ,Cast( Year(@DateInProcess ) as nVarchar(50) ) -- [YearName]
              )
    -- Add a day and loop again
    SET @DateInProcess = DateAdd(d, 1, @DateInProcess)
END
GO
--Checking: SELECT * FROM DimDates

 --=============== DimDepartments =================--

 INSERT INTO DimDepartments (DepartmentID
                                    ,DepartmentName)

 VALUES(100, 'DeptA'),
           (200, 'DeptB');

 SELECT DepartmentID
          ,DepartmentName
 FROM [dbo].[DimDepartments]

 -- SELECT * FROM [DWSampleData].[dbo].[DimDepartments]
```

```
--================= DimEmployees =================--

INSERT INTO DimEmployees (EmployeeID
                          ,EmployeeFullName
                          ,DepartmentKey
                          ,ManagerKey)

VALUES(100, 'Sue Jones', 1, 1),
      (200, 'Bob Smith', 1, 1),
      (300, 'Tim Thomas', 2, 1),
      (400, 'Jim Janus', 2, 3);

SELECT EmployeeID
      ,EmployeeFullName
      ,DepartmentKey
      ,ManagerKey
FROM [dbo].[DimEmployees]

--SELECT * FROM [DWSampleData].[dbo].[DimEmployees]

--================= DimProducts =================--

INSERT INTO DimProducts (ProductId
                         ,ProductName
                         ,ProductStandardPrice)

VALUES(100, 'ProdA', 9.9900),
      (200, 'ProdB', 19.9900),
      (300, 'ProdC', 9.9900);

SELECT ProductId
       ,ProductName
       ,ProductStandardPrice
FROM [dbo].[DimProducts]
GO
--SELECT * FROM [DWSampleData].[dbo].[DimProducts]
```

```
--================ FactInventories ================--

 INSERT INTO FactInventories (InventoryID
                             ,ProductKey
                             ,InventoryDateKey
                             ,InventoryCounts)

VALUES(1, 1, 20100101, 500),
      (1, 2, 20100101, 200),
      (1, 3, 20100201, 100),
      (2, 1, 20100201, 400),
      (2, 2, 20100201, 150),
      (2, 3, 20100201, 50);

SELECT InventoryID
      ,ProductKey
      ,InventoryDateKey
      ,InventoryCounts
FROM [dbo].[FactInventories]

--SELECT * FROM [DWSampleData].[dbo].[FactInventories]

--============ FactProductCategories ==============--

INSERT INTO FactProductCategories (ProductKey
                                  ,CategoryKey)

VALUES(1, 1),
      (2, 1),
      (3, 2);

SELECT ProductKey
      ,CategoryKey
FROM [dbo].[FactProductCategories]
--SELECT * FROM [DW205Final].[dbo].[FactProductCategories]
```

```
--================== FactSales ==================--

INSERT INTO FactSales (SalesID
                      ,SalesLineItemID
                      ,EmployeeKey
                      ,SalesDateKey
                      ,ShipDateKey
                      ,CustomerKey
                      ,ProductKey
                      ,SalesDollars
                      ,SalesUnits)

VALUES(1, 1, 2, 20100101, 20100102, 1, 1, 9.9900, 2),
    (1, 2, 2, 20100101, 20100102, 1, 3, 9.9900, 4),
          (2, 1, 4, 20100101, 20100102, 3, 1, 9.9900, 3),
          (3, 1, 4, 20100102, 20100103, 2, 1, 9.9900, 7),
          (3, 2, 4, 20100102, 20100103, 2, 3, 9.9900, 5),
          (3, 3, 4, 20100102, 20100103, 2, 2, 18.9900, 10),
          (4, 1, 2, 20100102, 20100103, 1, 3, 9.9900, 2),
          (5, 1, 2, 20100103, 20100104, 3, 2, 19.9900, 1),
          (5, 2, 2, 20100103, 20100104, 3, 3, 9.9900, 4),
          (6, 1, 2, 20100103, 20100104, 1, 2, 19.9900, 1),
          (6, 2, 2, 20100103, 20100104, 1, 3, 9.9900, 4),
          (6, 3, 2, 20100103, 20100104, 1, 3, 9.9900, 4);

SELECT SalesID
       ,SalesLineItemID
       ,EmployeeKey
       ,SalesDateKey
       ,ShipDateKey
       ,CustomerKey
       ,ProductKey
       ,SalesDollars
       ,SalesUnits
FROM [dbo].[FactSales]
GO
```

```
 /*Print out that the database was populated. */
 /*Convert to sysdatetime and then cast to varchar. */
 PRINT 'DWSampleData Filled Database: Populated All Data Successfully.'
 + CAST(CONVERT(varchar, SYSDATETIME(), 121) AS varchar (20))
 GO


/*************** DWBUSIT205Final ******************/
/*************** Tabular Modeling ******************/
/***************** Script Views ********************/

--Create views to speed up the data structure and not change anything.
--If the view exists, then drop it first.
IF EXISTS(SELECT 1 FROM sys.views WHERE NAME='vDimCustomers' and TYPE='v')
DROP VIEW vDimCustomers;
GO
CREATE VIEW vDimCustomers
AS
/*
Created By: Chris Singleton
Date: 06/05/2017
Objective: View of the entire DimCustomers Table.
*/
SELECT [CustomerKey]
            ,[CustomerID]
            ,CAST([CustomerName] AS NVARCHAR(50)) AS CustomerName
            ,CAST([CustomerCity] AS NVARCHAR(50)) AS CustomerCity
            ,CAST([CustomerState] AS NVARCHAR(50)) AS CustomerState
FROM [DWSampleData].dbo.DimCustomers
GO

--If the view exists, then drop it first.
IF EXISTS(SELECT 1 FROM sys.views WHERE NAME='vDimDates' and TYPE='v')
DROP VIEW vDimDates;
GO
```

```sql
CREATE VIEW vDimDates
AS
/*
Created By: Chris Singleton
Date: 06/05/2017
Objective: View of the entire DimDates Table.
*/
--SELECT
SELECT [DateKey]
      ,CAST(CAST(DateKey AS char(8)) AS Date) AS [Date]
            ,[DateName]
            ,[MonthKey]
            ,[MonthName]
            ,[YearKey]
            ,[YearName]
FROM [DWSampleData].[dbo].[DimDates]
GO

--If the view exists, then drop it first.
IF EXISTS(SELECT 1 FROM sys.views WHERE NAME='vDimProducts' and TYPE='v')
DROP VIEW vDimProducts;
GO
CREATE VIEW vDimProducts
AS
/*
Created By: Chris Singleton
Date: 06/05/2017
Objective: View of the entire DimProducts Table.
*/
SELECT [ProductKey]
            ,[ProductID]
            ,CAST([ProductName] AS NVARCHAR(50)) AS ProductName
            ,CONVERT(DECIMAL(7,2),[ProductStandardPrice]) AS ProductStandardPrice
FROM [DWSampleData].[dbo].[DimProducts]
GO
```

```sql
--If the view exists, then drop it first.
IF EXISTS(SELECT 1 FROM sys.views WHERE NAME='vDimCategories' and TYPE='v')
DROP VIEW vDimCategories;
GO
CREATE VIEW vDimCategories
AS
/*
Created By: Chris Singleton
Date: 06/05/2017
Objective: View of the entire DimCategories Table.
*/
SELECT [CategoryId]
      ,CAST([CategoryName] AS NVARCHAR(50)) AS CategoryName
FROM [DWSampleData].[dbo].[DimCategories]
GO

--If the view exists, then drop it first.
IF EXISTS(SELECT 1 FROM sys.views WHERE NAME='vDimDeparments' and TYPE='v')
DROP VIEW vDimDeparments;
GO
CREATE VIEW vDimDeparments
AS
/*
Created By: Chris Singleton
Date: 06/05/2017
Objective: View of the entire DimDepartments Table.
*/
SELECT [DepartmentKey]
      ,[DepartmentID]
      ,[DepartmentName]
FROM [DWSampleData].[dbo].[DimDepartments]
GO
```

```sql
--If the view exists, then drop it first.
IF EXISTS(SELECT 1 FROM sys.views WHERE NAME='vDimEmployees' and TYPE='v')
DROP VIEW vDimEmployees;
GO

CREATE VIEW vDimEmployees
AS
/*
Created By: Chris Singleton
Date: 06/05/2017
Objective: View of the entire DimEmployees Table.
*/
SELECT [EmployeeKey]
      ,[EmployeeID]
      ,CAST([EmployeeFullName] AS NVARCHAR(50)) AS EmployeeFullName
      ,[DepartmentKey]
      ,[ManagerKey]
FROM [DWSampleData].[dbo].[DimEmployees]
GO

--If the view exists, then drop it first.
IF EXISTS(SELECT 1 FROM sys.views WHERE NAME='vFactInventories' and TYPE='v')
DROP VIEW vFactInventories;
GO
CREATE VIEW vFactInventories
AS
/*
Created By: Chris Singleton
Date: 06/05/2017
Objective: View of the entire FactInventories Table.
*/
SELECT [InventoryID]
      ,[ProductKey]
      ,[InventoryDateKey]
      ,CAST(CAST(InventoryDateKey AS char(8)) AS Date) AS [InventoryDate]
      ,[InventoryCounts]
FROM [DWSampleData].[dbo].[FactInventories]
GO
```

```sql
--If the view exists, then drop it first.
IF EXISTS(SELECT 1 FROM sys.views WHERE NAME='vFactProductCategories' and TYPE='v')
DROP VIEW vFactProductCategories;
GO
CREATE VIEW vFactProductCategories
AS
/*
Created By: Chris Singleton
Date: 06/05/2017
Objective: View of the entire FactProductCategories Table.
*/
SELECT [ProductKey]
      ,[CategoryKey]
FROM [DWSampleData].[dbo].[FactProductCategories]
GO
--If the view exists, then drop it first.
IF EXISTS(SELECT 1 FROM sys.views WHERE NAME='vFactSales' and TYPE='v')
DROP VIEW vFactSales;
GO
CREATE VIEW vFactSales
AS
/*
Created By: Chris Singleton
Date: 06/05/2017
Objective: View of the entire FactSales Table.
*/
SELECT [SalesID]
      ,[SalesLineItemID]
      ,[EmployeeKey]
      ,[SalesDateKey]
      ,CAST(CAST(SalesDateKey AS char(8)) AS Date) AS [SalesDate]
      ,[ShipDateKey]
      ,CAST(CAST(ShipDateKey AS char(8)) AS Date) AS [ShipDate]
      ,[CustomerKey]
      ,[ProductKey]
      ,CONVERT(DECIMAL(7,2),[SalesDollars]) AS SalesDollars
      ,[SalesUnits]
FROM [DWSampleData].[dbo].[FactSales]
```

```
GO

/*Print out that the Views were created successfully */
/*Convert to sysdatetime and then cast to varchar. */
PRINT 'DWSampleData Views: Created All Views Successfully.'
+ CAST(CONVERT(varchar, SYSDATETIME(), 121) AS varchar (20))

GO

EXEC sp_help DimCustomers
EXEC sp_help DimDates
EXEC sp_help DimDepartments
EXEC sp_help DimEmployees
EXEC sp_help DimProducts
EXEC sp_help FactInventories
EXEC sp_help FactProductCategories
EXEC sp_help FactSales
```