**CIS 310-02**

**Assignment 8, Due 11/1/17**

**Jessica Hoffman**

--Jessica Hoffman

--CIS 310-02

--Assignment 8

--Due 11/1/17

--1

SELECT ItemID, Description, ListPrice

FROM PET..Merchandise

WHERE ListPrice > (SELECT AVG(ListPrice)

FROM PET..Merchandise)

--2

CREATE VIEW AVERAGECOST AS (

(SELECT ItemID, AVG(Cost) AS 'AVERAGE COST'

FROM Pet..OrderItem

GROUP BY ItemID)

)

CREATE VIEW AVERAGEPRICE AS (

(SELECT ItemID, AVG(SalePrice) AS 'AVERAGE SALE PRICE'

FROM Pet..SaleItem

GROUP BY ItemID)

)

SELECT C.ItemID AS ITEMID, C.[AVERAGE COST], P.[AVERAGE SALE PRICE]

FROM AVERAGECOST C INNER JOIN AVERAGEPRICE P ON P.ItemID = C.ItemID

WHERE P.[AVERAGE SALE PRICE] > 1.5\*C.[AVERAGE COST]

--3

CREATE VIEW TOTSALES AS

SELECT S.EmployeeID AS EMPLOYEEID, SUM(SI.SalePrice) AS TOTALSALES

FROM Pet..SaleItem SI INNER JOIN Pet..Sale S ON SI.SaleID = S.SaleID

GROUP BY S.EmployeeID

CREATE VIEW PCTOFTOTAL AS

SELECT S.EmployeeID AS EMPLOYEEID, SUM(SI.SalePrice)\*100/SUM(TOTALSALES) AS PCTSALES

FROM TOTSALES, Pet..SaleItem SI INNER JOIN Pet..Sale S ON SI.SaleID = S.SaleID

GROUP BY S.EmployeeID

SELECT T.EMPLOYEEID,

E.LastName AS LASTNAME,

T.TOTALSALES,

P.PCTSALES

FROM PET..Employee E INNER JOIN TOTSALES T ON T.EMPLOYEEID = E.EmployeeID

INNER JOIN PCTOFTOTAL P ON P.EMPLOYEEID = E.EmployeeID

--4

CREATE VIEW SHIPPINGCOSTS AS (

SELECT M.SupplierID,

M.PONumber,

M.ShippingCost,

SUM(O.Cost \* O.Quantity) AS Total

FROM Pet..MerchandiseOrder M INNER JOIN Pet..OrderItem O ON O.PONumber = M.PONumber

GROUP BY M.PONumber, M.SupplierID, M.ShippingCost

)

CREATE VIEW AVGSHIPCOSTS AS (

SELECT S.SupplierID,

S.Name,

AVG(SC.ShippingCost/SC.Total) AS AvgPctShipCost

FROM Pet..Supplier S INNER JOIN SHIPPINGCOSTS SC ON SC.SupplierID = S.SupplierID

GROUP BY S.SupplierID, S.Name

)

SELECT TOP(1) A.SupplierID, A.Name, A.AvgPctShipCost

FROM AVGSHIPCOSTS A

ORDER BY A.AvgPctShipCost DESC

--5

CREATE VIEW ANIMALSPENTTOTAL AS

SELECT S.CustomerID AS CustomerID, SUM(A.SalePrice) AS TOTSPENT

FROM Pet..Sale S INNER JOIN Pet..SaleAnimal A ON A.SaleID = S.SaleID

GROUP BY S.CustomerID

CREATE VIEW MERCHSPENTTOTAL AS

SELECT S.CustomerID AS CustomerID, SUM(I.SalePrice) AS TOTSPENT

FROM Pet..Sale S INNER JOIN Pet..SaleItem I ON I.SaleID = S.SaleID

GROUP BY S.CustomerID

SELECT TOP(1) C.CustomerID,

C.LastName,

C.FirstName,

A.TOTSPENT AS Total,

M.TOTSPENT AS Total,

(A.TOTSPENT + M.TOTSPENT) AS GrandTotal

FROM Pet..Customer C INNER JOIN ANIMALSPENTTOTAL A ON C.CustomerID = A.CustomerID

INNER JOIN MERCHSPENTTOTAL M ON A.CustomerID = M.CustomerID

ORDER BY GrandTotal DESC

--6

CREATE VIEW OCTOBERTOTAL AS

SELECT S.CustomerID AS CustomerID,

SUM(SI.SalePrice) AS OctTotal

FROM Pet..MerchandiseOrder O INNER JOIN Pet..OrderItem I ON O.PONumber = I.PONumber

INNER JOIN Pet..SaleItem SI ON SI.ItemID = I.ItemID

INNER JOIN Pet..Sale S ON S.SaleID = SI.SaleID

WHERE MONTH(O.ORDERDATE) = 10

GROUP BY S.CustomerID

CREATE VIEW MAYTOTAL AS

SELECT S.CustomerID AS CustomerID,

SUM(SI.SalePrice) AS MayTot

FROM Pet..MerchandiseOrder O INNER JOIN Pet..OrderItem I ON O.PONumber = I.PONumber

INNER JOIN Pet..SaleItem SI ON SI.ItemID = I.ItemID

INNER JOIN Pet..Sale S ON S.SaleID = SI.SaleID

WHERE MONTH(O.ORDERDATE) = 5

GROUP BY S.CustomerID

SELECT C.CustomerID, C.LastName, C.FirstName, M.MayTot AS MayTotal

FROM Pet..Customer C INNER JOIN OCTOBERTOTAL O ON O.CustomerID = C.CustomerID

INNER JOIN MAYTOTAL M ON M.CustomerID = C.CustomerID

WHERE M.MayTot > 100 AND O.OctTotal > 50

--7

CREATE VIEW DOGFOODJANJULY AS (

SELECT O.ItemID,

M.Description,

SUM(O.Quantity) AS Purchased,

SUM(S.Quantity) AS Sold

FROM Pet..Merchandise M INNER JOIN Pet..OrderItem O ON O.ItemID = M.ItemID

INNER JOIN Pet..SaleItem S ON O.ItemID = S.ItemID

INNER JOIN Pet..MerchandiseOrder MO ON MO.PONumber = O.PONumber

WHERE (MONTH(MO.OrderDate) BETWEEN 1 AND 7) AND M.Description = 'Dog Food-Can-Premium'

GROUP BY O.ItemID, M.Description

)

SELECT D.Description, D.ItemID, D.Purchased, D.Sold, (D.Purchased - D.Sold) AS NetIncrease

FROM DOGFOODJANJULY D INNER JOIN Pet..Merchandise M ON M.ItemID = D.ItemID

--8

CREATE VIEW NONJULYSALES AS

(

SELECT M.ItemID,

M.Description,

M.ListPrice

FROM Pet..Merchandise M INNER JOIN Pet..SaleItem SI ON SI.ItemID = M.ItemID

INNER JOIN Pet..Sale S ON S.SaleID = SI.SaleID

WHERE SI.ItemID NOT IN (SELECT SI.ItemID

FROM Pet..Sale S

INNER JOIN Pet..SaleItem SI ON SI.SaleID = S.SaleID

WHERE MONTH(S.SaleDate) = 7)

GROUP BY M.ItemID, M.Description, M.ListPrice

)

SELECT \*

FROM NONJULYSALES

WHERE ListPrice > 50

--9

CREATE VIEW MERCHOVER100UNITS AS

SELECT M.ItemID AS ItemID,

M.Description AS Description,

M.QuantityOnHand AS QuantityOnHand

FROM Pet..Merchandise M

WHERE M.QuantityOnHand > 100

SELECT M.ItemID, M.Description, M.QuantityOnHand, O.ItemID

FROM MERCHOVER100UNITS M INNER JOIN Pet..OrderItem O ON M.ItemID = O.ItemID

LEFT OUTER JOIN Pet..MerchandiseOrder MO ON MO.PONumber = O.PONumber

WHERE YEAR(MO.OrderDate) != 2004

--10

SELECT M.ItemID, M.Description, M.QuantityOnHand, O.ItemID

FROM MERCHOVER100UNITS M INNER JOIN Pet..OrderItem O ON O.ItemID = M.ItemID

WHERE EXISTS (SELECT MO.OrderDate, O.ItemID

FROM Pet..MerchandiseOrder MO

INNER JOIN Pet..OrderItem O ON O.PONumber = MO.PONumber

WHERE YEAR(MO.OrderDate) != 2004)

--11

CREATE TABLE Category (

Category varchar(10) NOT NULL,

Low int,

High int

)

INSERT INTO Category (

Category,

Low,

High

) VALUES

('Weak',

0,

200),

('Good',

200,

800),

('Best',

800,

10000)

CREATE VIEW CustomerGrandTotal AS (

SELECT S.CustomerID,

(SUM(A.SalePrice) + SUM(M.SalePrice)) AS GrandTotal

FROM Pet..Sale S INNER JOIN Pet..SaleAnimal A ON A.SaleID = S.SaleID

INNER JOIN Pet..SaleItem M ON M.SaleID = S.SaleID

GROUP BY S.CustomerID

)

CREATE VIEW CustomerCategory AS (

SELECT CA.Category, CGT.CustomerID

FROM Category CA, CustomerGrandTotal CGT

WHERE CGT.GrandTotal >= CA.Low AND CGT.GrandTotal < CA.High OR CGT.GrandTotal > 10000

)

SELECT C.CustomerID,

C.LastName,

C.FirstName,

CGT.GrandTotal,

CC.Category

FROM CustomerGrandTotal CGT INNER JOIN Pet..Customer C ON C.CustomerID = CGT.CustomerID

INNER JOIN CustomerCategory CC ON CC.CustomerID = C.CustomerID

--12

CREATE VIEW SUPPLIERSJUNE AS (

SELECT S.SupplierID,

S.Name AS Name,

MO.OrderDate,

'Merchandise' AS OrderType

FROM Pet..Supplier S INNER JOIN Pet..MerchandiseOrder MO ON MO.SupplierID = S.SupplierID

WHERE MONTH(MO.OrderDate) = 6

UNION ALL

SELECT S.SupplierID,

S.Name AS Name,

AO.OrderDate,

'Animal' AS OrderType

FROM Pet..Supplier S INNER JOIN Pet..AnimalOrder AO ON AO.SupplierID = S.SupplierID

WHERE MONTH(AO.OrderDate) = 6

)

SELECT SJ.Name, SJ.OrderType

FROM SUPPLIERSJUNE SJ

--13

DROP TABLE Category

CREATE TABLE Category (

Category varchar(10) NOT NULL,

Low int,

High int

)

--14

CREATE VIEW CategoryTbl AS (

SELECT Category, Low, High

FROM Category

)

INSERT INTO CategoryTbl

VALUES (

'Weak',

0,

200

)

--15

UPDATE CategoryTbl

SET High = 400

WHERE Category = 'Weak'

--17

DELETE TOP (1)

FROM CategoryTbl

--18

CREATE VIEW EmployeeTemp AS (

SELECT \*

FROM Pet..Employee

);

DELETE FROM EmployeeTemp

ALTER VIEW EmployeeTemp AS (

SELECT \*

FROM Pet..Employee

)