```
--ASSIGMENT A8


--1

--List the products with a list price greater than the average list price of all products.

SELECT  ItemID, Description, ListPrice

FROM   CIS310A8..Merchandise

WHERE ListPrice >

                (SELECT AVG(ListPrice)

                FROM   CIS310A8..Merchandise)

ORDER BY ListPrice


--2

--Which merchandise items have an average sale price more than 50 percent higher than their

--average purchase cost?

SELECT  M.ItemID, AVG(OI.Cost) AS AverageCost, AVG(SI.SalePrice) AS AverageSalePrice

FROM   CIS310A8..Merchandise M INNER JOIN CIS310A8..OrderItem OI ON OI.ItemID = M.ItemID

                INNER JOIN CIS310A8..SaleItem SI ON M.ItemID = SI.ItemID

GROUP BY M.ItemID

HAVING AVG(SI.SalePrice) > AVG(OI.Cost)*1.5


--3

--List the employees and their total merchandise sales expressed as a percentage of

--total merchandise sales for all employees.

CREATE VIEW TOTAL_EMPLOYEE_SALES AS

SELECT  E.EmployeeID, E.LastName,
SUM((SI.SalePrice*SI.Quantity*S.SalesTax)+(SI.SalePrice*SI.Quantity)) AS TotalSales

FROM   CIS310A8..SaleItem SI INNER JOIN CIS310A8..Sale S ON SI.SaleID = S.SaleID

                INNER JOIN CIS310A8..Employee E ON E.EmployeeID = S.EmployeeID

GROUP BY E.EmployeeID, E.LastName
```

```
SELECT  *, (TotalSales/(SELECT SUM(TotalSales) FROM    TOTAL_EMPLOYEE_SALES))*100 AS PctSales
FROM   TOTAL_EMPLOYEE_SALES
ORDER BY EmployeeID

--4
--On average, which supplier charges the highest shipping cost as a percent of the
--merchandise order total?
CREATE VIEW PERCENT_COST AS
SELECT  SUP.SupplierID, SUP.Name,
AVG(MO.ShippingCost/(OI.Cost*OI.Quantity+MO.ShippingCost))*100 AS PctShipCostTotal
FROM   CIS310A8..Supplier SUP INNER JOIN CIS310A8..MerchandiseOrder MO ON SUP.SupplierID =
MO.SupplierID
                INNER JOIN CIS310A8..OrderItem OI ON OI.PONumber = MO.PONumber
GROUP BY SUP.SupplierID, SUP.Name

SELECT  SUP.SupplierID, SUP.Name, AVG(MO.ShippingCost/OI.Cost) AS PctShipCost
FROM   CIS310A8..Supplier SUP INNER JOIN CIS310A8..MerchandiseOrder MO ON SUP.SupplierID =
MO.SupplierID
                INNER JOIN CIS310A8..OrderItem OI ON OI.PONumber = MO.PONumber
GROUP BY SUP.SupplierID, SUP.Name
HAVING          AVG(MO.ShippingCost/OI.Cost) = (SELECT MAX(PctShipCostTotal)
                                                FROM PERCENT_COST)

--5
--Which customer has given us the most total money for animals and merchandise?
ALTER VIEW ANIMAL_PURCHASE_TOTALS AS
SELECT  C.CustomerID, C.LastName, C.FirstName, SUM((SA.SalePrice)*(1+S.SalesTax)) AS AnimalTotal
FROM   CIS310A8..Customer C INNER JOIN CIS310A8..Sale S ON C.CustomerID = S.CustomerID
```

```
                INNER JOIN CIS310A8..SaleAnimal SA ON S.SaleID = SA.SaleID
GROUP BY C.CustomerID, C.LastName, C.FirstName

ALTER VIEW MERCH_PURCHASE_TOTALS AS
SELECT  C.CustomerID, C.LastName, C.FirstName, SUM((SI.SalePrice*SI.Quantity)*(1+S.SalesTax)) AS
MerchTotal
FROM    CIS310A8..Customer C INNER JOIN CIS310A8..Sale S ON C.CustomerID = S.CustomerID
                INNER JOIN CIS310A8..SaleItem SI ON S.SaleID = SI.SaleID
GROUP BY C.CustomerID, C.LastName, C.FirstName

ALTER VIEW ANIMAL_AND_MERCH_TOTALS AS
SELECT  C.CustomerID, C.LastName, C.FirstName, MPT.MerchTotal, APT.AnimalTotal,
SUM(MPT.MerchTotal + APT.AnimalTotal) AS GrandTotal
FROM    ANIMAL_PURCHASE_TOTALS APT INNER JOIN MERCH_PURCHASE_TOTALS MPT ON
APT.CustomerID = MPT.CustomerID
                INNER JOIN CIS310A8..Customer C ON C.CustomerID = MPT.CustomerID
GROUP BY C.CustomerID, C.LastName, C.FirstName, MPT.MerchTotal, APT.AnimalTotal

SELECT  *
FROM    ANIMAL_AND_MERCH_TOTALS
WHERE GrandTotal =
                (SELECTMAX(GrandTotal)
                FROM    ANIMAL_AND_MERCH_TOTALS)

--6
--Which customers who bought more than $100 in merchandise in May also spent more
--than $50 on merchandise in October?
SELECT  C.CustomerID, C.LastName, C.FirstName, SUM(SI.SalePrice*SI.Quantity) AS MayTotal
FROM    CIS310A8..Customer C INNER JOIN CIS310A8..Sale S ON C.CustomerID = S.CustomerID
                INNER JOIN CIS310A8..SaleItem SI ON S.SaleID = SI.SaleID
```

```
WHERE C.CustomerID IN
                    (
                    SELECT  C.CustomerID
                    FROM    CIS310A8..Customer C INNER JOIN CIS310A8..Sale S ON C.CustomerID =
S.CustomerID
                                INNER JOIN CIS310A8..SaleItem SI ON S.SaleID = SI.SaleID
                    WHERE C.CustomerID = S.CustomerID AND S.SaleDate LIKE '%OCT%'
                    GROUP BY C.CustomerID, C.LastName, C.FirstName, S.SaleDate
                    HAVING          SUM(SI.SalePrice*SI.Quantity) > 50
                    )
AND         S.SaleDate LIKE '%MAY%'
GROUP BY C.CustomerID, C.LastName,C.FirstName
HAVING SUM(SI.SalePrice*SI.Quantity) > 100

--7
--What was the net change in quantity on hand for premium canned dog food between
--January 1 and July 1?
SELECT  M.Description, M.ItemID, SUM(OI.Quantity) AS Purchased, SUM(SI.Quantity) AS Sold,
                                        SUM(OI.Quantity - SI.Quantity) AS NetIncrease
FROM    CIS310A8..OrderItem OI INNER JOIN CIS310A8..Merchandise M ON OI.ItemID = M.ItemID
                    INNER JOIN CIS310A8..SaleItem SI ON M.ItemID = SI.ItemID
                    INNER JOIN CIS310A8..Sale S ON SI.SaleID = S.SaleID
WHERE M.ItemID = 16 AND S.SaleDate BETWEEN '1-JAN-2004' AND '1-JULY-2004'
GROUP BY M.Description, M.ItemID

--8
--Which merchandise items with a list price of more than $50 had no sales in July?
SELECT  M.ItemID, M.Description, M.ListPrice
FROM    CIS310A8..Merchandise M INNER JOIN CIS310A8..SaleItem SI ON SI.ItemID = M.ItemID
```

```
                INNER JOIN CIS310A8..Sale S ON SI.SaleID = S.SaleID
WHERE M.ListPrice > 50 AND S.SaleDate NOT BETWEEN '2004-7-1' AND '2004-7-31'
GROUP BY M.ItemID, M.Description, M.ListPrice

--9
--Which merchandise items with more than 100 units on hand have not been ordered in 2004?
--Use an outer join to answer the question.
SELECT  M.ItemID, M.Description, M.QuantityOnHand
FROM   CIS310A8..MerchandiseOrder MO LEFT OUTER JOIN CIS310A8..OrderItem OI ON MO.PONumber
= OI.PONumber
                LEFT OUTER JOIN CIS310A8..Merchandise M ON OI.ItemID = M.ItemID
WHERE M.QuantityOnHand > 100 AND NOT YEAR(MO.PONumber) = 2004
GROUP BY M.ItemID, M.Description, M.QuantityOnHand

--10
--Which merchandise items with more than 100 units on hand have not been ordered in 2004?
--Use a subquery to answer the question.
SELECT  M.ItemID, M.Description, M.QuantityOnHand
FROM   CIS310A8..Merchandise M
WHERE M.QuantityOnHand  > 100 AND M.ItemID NOT IN
                (
                        SELECT  OI.ItemID
                        FROM   CIS310A8..OrderItem OI
                        WHERE OI.PONumber IN
                                (
                                SELECT  MO.PONumber
                                FROM   CIS310A8..MerchandiseOrder MO
                                WHERE YEAR(MO.OrderDate) = 2004
                                )
```

```
                    )

--11
--Save a query to answer Exercise 5: total amount of money spent by each customer.
--Create the table shown to categorize customers based on sales. Write a query that lists
--each customer from the first query and displays the proper label. Must use only SQL statements
--and include all statements used in the proper order.
CREATE VIEW ANIMAL_AMOUNT_BY_CUSTOMER AS
SELECT  C.CustomerID, C.LastName, C.FirstName, ISNULL(SUM((SA.SalePrice)*(1+S.SalesTax)),0) AS
AnimalTotal
FROM    CIS310A8..Customer C INNER JOIN CIS310A8..Sale S ON C.CustomerID = S.CustomerID
                    FULL OUTER JOIN CIS310A8..SaleAnimal SA ON S.SaleID = SA.SaleID
GROUP BY C.CustomerID, C.LastName, C.FirstName

CREATE VIEW MERCH_AMOUNT_BY_CUSTOMER AS
SELECT  C.CustomerID, C.LastName, C.FirstName,
ISNULL(SUM((SI.SalePrice*SI.Quantity)*(1+S.SalesTax)),0) AS MerchTotal
FROM    CIS310A8..Customer C INNER JOIN CIS310A8..Sale S ON C.CustomerID = S.CustomerID
                    FULL OUTER JOIN CIS310A8..SaleItem SI ON S.SaleID = SI.SaleID
GROUP BY C.CustomerID, C.LastName, C.FirstName

CREATE VIEW TOTAL_AMOUNT_BY_CUSTOMER AS
SELECT  C.CustomerID, C.LastName, C.FirstName, ISNULL(SUM(MABC.MerchTotal +
AABC.AnimalTotal),0) AS GrandTotal
FROM    ANIMAL_AMOUNT_BY_CUSTOMER AABC INNER JOIN MERCH_AMOUNT_BY_CUSTOMER MABC
ON AABC.CustomerID = MABC.CustomerID
                    FULL OUTER JOIN CIS310A8..Customer C ON C.CustomerID = MABC.CustomerID
GROUP BY C.CustomerID, C.LastName, C.FirstName
```

```
CREATE TABLE Customer_Category
(
	Category	varchar(15)	NOT NULL,
	Low			money	NOT NULL,
	High		money	NOT NULL,
)

INSERT INTO Customer_Category
(Category, Low, High) values ('Weak', 0, 200);
INSERT INTO Customer_Category
(Category, Low, High) values ('Good', 200, 800);
INSERT INTO Customer_Category
(Category, Low, High) values ('Best', 800, 10000);
INSERT INTO Customer_Category
(Category, Low, High) values ('Above Best', 10000, 30000);

CREATE TABLE CUSTOMER_SPEND_RANK
(
	CustomerID	int NOT NULL,
	LastName	varchar (15),
	FirstName	varchar(15),
	GrandTotal	money,
	Category	varchar(15),
)

INSERT INTO CUSTOMER_SPEND_RANK (CustomerID, LastName, FirstName, GrandTotal, Category)
SELECT  CustomerID, LastName, FirstName, GrandTotal, 'EMPTY'
FROM   TOTAL_AMOUNT_BY_CUSTOMER TABC
GROUP BY TABC.CustomerID, TABC.LastName, TABC.FirstName, TABC.GrandTotal
```

```
UPDATE CUSTOMER_SPEND_RANK
SET Category = 'Weak'
WHERE GrandTotal BETWEEN 0 AND 200

UPDATE CUSTOMER_SPEND_RANK
SET Category = 'Good'
WHERE GrandTotal BETWEEN 200 AND 800

UPDATE CUSTOMER_SPEND_RANK
SET Category = 'Best'
WHERE GrandTotal BETWEEN 800 AND 10000

UPDATE          CUSTOMER_SPEND_RANK
SET Category = 'Above Best'
WHERE GrandTotal BETWEEN 10000 AND 30000

--12
--List all suppliers(animals and merchandise) who sold us items in June. Identify whether
--they sold us animals or merchandise.
SELECT  SUP.Name AS SupplierName, 'Merchandise' AS OrderType
FROM   CIS310A8..Supplier SUP INNER JOIN CIS310A8..MerchandiseOrder MO ON SUP.SupplierID =
MO.SupplierID
                INNER JOIN CIS310A8..OrderItem OI ON MO.PONumber = OI.PONumber
GROUP BY SUP.Name, SUP.SupplierID, OI.Cost
HAVING          OI.Cost > 0 AND
                SUP.SupplierID IN
                (
                 SELECT MO.SupplierID
```

```
                    FROM   CIS310A8..MerchandiseOrder MO
                    WHERE  OrderDate BETWEEN '2004-6-1' AND '2004-7-1'
                    GROUP BY MO.SupplierID
                    )
UNION

SELECT  SUP.Name AS SupplierName, 'Animal' AS OrderType
FROM   CIS310A8..Supplier SUP INNER JOIN CIS310A8..AnimalOrder AO ON SUP.SupplierID =
AO.SupplierID
                    INNER JOIN CIS310A8..AnimalOrderItem AOI ON AOI.OrderID = AOI.OrderID
GROUP BY SUP.Name, SUP.SupplierID, AOI.Cost
HAVING AOI.Cost > 0 AND
                    SUP.SupplierID IN
                    (
                    SELECT AO.SupplierID
                    FROM  CIS310A8..AnimalOrder AO
                    WHERE  AO.OrderDate BETWEEN '2004-6-1' AND '2004-7-1'
                    GROUP BY AO.SupplierID
                    )

--13
--Write a query to change the High Value to 400 in the first row of the table
--CATEGORYTABLE (Customer_Category).
UPDATE          Customer_Category
SET High = 400
WHERE Category = 'Weak'

--14
--Drop the table Category.
```

```
DROP TABLE CIS310A8..Category


--15
--Create a query to delete the first row of the table CATEGORYTABLE (Customer_Category).
DELETE
FROM   Customer_Category
WHERE Category LIKE '%Weak%'



--16
--Create a copy of the Employee table structure. Use a delete query to remove
--all data from the copy. Write a  query to copy from the original employee table
--into the new one.
SELECT  *
INTO     EMPLOYEE_DUPLICATE
FROM   CIS310A8..Employee

DELETE
FROM   EMPLOYEE_DUPLICATE

INSERT
INTO EMPLOYEE_DUPLICATE
SELECT  *
FROM   CIS310A8..Employee
```