**Assignment 2**

**Vaidehi Gosawi 662635635**

**2. (15%) Compose a relational algebra expression that would show each product id and product name that uses all of raw materials with material description = cherry [hint: use a Division operator].**

**Ans 2.** Product\_ID, Product\_Name((πProduct\_ID,Material\_ID(Uses\_t)/(πMaterial\_ID(σMaterial\_description=’Cherry’ ) Raw\_materials\_t)) X Product\_t)

**3. (20%) Compose a SQL statement that is equivalent to Question 1 above. Note, you might want to try and execute this SQL statement against the NewPVF database in SQL Server to see if it works as intended.**

**Ans 3.** Select Vendor\_name

from Vendor\_t v

join Supplies\_t s on v.Vendor\_ID= s.Vendor\_ID

where Material\_ID IN (select r.Material\_ID

from Raw\_Materials\_t r

join Supplies\_t s

on r.Material\_ID = s.Material\_ID

where r.Material\_description= 'Walnut' AND r.Unit\_Price < 14)

**4. (20%) Compose a SQL statement that is equivalent to Question 2 above. Note, you might want to try and execute this SQL statement against the NewPVF database in SQL Server to see if it works as intended [hint: use a correlated subquery].**

**Ans.**

Select product\_id, product\_name

From product\_t as p

Where not exists ( Select r.material\_id

From raw\_materials\_t as r

Where material\_description = 'Cherry'

Except

Select u.material\_id

From uses\_t as u

Where u.product\_id = p.product\_id)

**5. (20%) Compose an SQL statement to generate a list of two least expensive vendors (suppliers) for each raw material. In the result table, show the following columns: material ID, material description, vendor ID, vendor name, and the supplier's unit price. Sort the result table by material ID and supplier’s unit price in ascending order. Note: If a raw material has only one vendor (supplier), that supplier and its unit price for the raw material should also be in the result (output) table [hint: use a correlated subquery].**

**Ans 5.** SELECT s.Material\_ID, rm.Material\_Description, v.Vendor\_ID, v.Vendor\_name, s.Unit\_price

FROM (Supplies\_t s INNER JOIN Raw\_Materials\_t rm ON s.Material\_ID = rm.Material\_ID )

INNER JOIN Vendor\_t v ON v.Vendor\_ID = s.Vendor\_ID

WHERE s.Unit\_price IN (SELECT TOP 2 s2.Unit\_price

FROM Supplies\_t s

WHERE s.Material\_ID = R.Material\_ID

ORDER BY s2.Material\_ID ASC, s2.Unit\_price ASC );

**6. (15%) Compose a query that would find customer(s) with exactly one order in October 2011. In the result table, display Customer\_ID, Customer\_Name, Order\_ID, and Order\_Date.**

**Ans 6.** Select c.Customer\_ID, c.Customer\_Name, o.Order\_ID, o.Order\_Date

from Customer\_t c

join Order\_t o on c.Customer\_ID= o.Customer\_ID

where c.Customer\_ID IN (Select Customer\_ID

from Order\_t

where Order\_Date BETWEEN '2011-10-01' AND '2011-10-31'

group by Customer\_ID

having COUNT(\*)=1) AND Order\_Date BETWEEN '2011-10-01' AND '2011-10-31' ;