Lab 06

On sales database

1. Compute total sales of a given item (say item code=1103).

$$\mathcal{F}_{sum(qty*rate)}\,(\sigma_{itemcode='1103'}\,(INVOICEDETAILS))$$

select sum(qty) from invoicedetails where itemcode='1103';

2. What is sale for a given date?
   [Total sale amount for a give date; it can be computed by summing (qty*rate) from InvoiceDetails of invoices on given date].

$$\mathcal{F}_{sum(id.qty*id.rate)}\,(\sigma_{i.invdate='2010-06-30'}\,(INVOICEDETAILS\ id)$$

$$\bowtie\,(INVOICE\ i))$$

select sum (id.qty*id.rate) from invoicedetails id natural join invoice i where i.invdate = '2010-06-30';

3. List item codes of top 3 most sold item based on quantity.

$$\pi_{\text{itemcode}}(\sigma_{\text{ORDER BY qty desc LIMIT 3}}(\text{INVOICEDETAILS}))$$

select itemcode from invoicedetails order by qty desc limit 3;

4. List (item codes, item name, category) of top 3 most sold item based on quantity.

$$\pi_{\text{i.itemcode, i.name, i.category}}(\sigma_{\text{ORDER BY id.qty desc LIMIT 3}}(\text{ITEMS i}$$
$$\bowtie_{\text{i.code = id.itemcode}} \text{INVOICEDETAILS id}))$$

select i.code,i.name,i.category from items i inner join invoicedetails id on i.code=id.itemcode order by id.qty desc limit 3;

5. Most valuable customer (customer id) in terms of purchase values. Customer that sums of maximum sale amount.

$$_{\text{c.custid, i.invno}}\mathcal{F}_{\text{SUM(id.rate)} \rightarrow \text{rate}}(\sigma_{\text{ORDER BY id.qty desc LIMIT}}$$
$$_3(\text{CUSTOMER c} \bowtie_{\text{c.custid = i.customerid}} \text{INVOICE} \bowtie$$
$$\text{INVOICEDETAILS id}))$$

select c.custid, sum(id.rate) as rate from customer c inner join invoice i on c.custid=i.customerid natural join invoicedetails id group by c.custid, i.invno order by rate desc limit 1;

OR

select customerid from invoice where invno = (select invno from invoicedetails where rate = (select max(rate) from invoicedetails)) ;

6. Most valuable customer (customer id) in terms profit to the company. Assume that profit on an item sale can be computed by formula:
Rate (from invoicedetails relation) – AveragePurchasePrice

$$_{c.custid}\mathcal{F}_{\text{SUM(id.rate-it.averagepurchaseprice)} \rightarrow \text{profit}} \left(\sigma_{\text{ORDER BY profit desc LIMIT 1}} (\text{INVOICE i} \bowtie \text{INVOICEDETAILS id} \bowtie_{\text{id.itemcode=it.icode}} \text{ITEMS it}) \right)$$

OR

$_{c.custid,\ i.invno}\mathcal{F}_{SUM(id.rate)\ \rightarrow\ profit}\big(\sigma_{ORDER\ BY\ profit\ desc\ LIMIT}$

$_1(CUSTOMER\ c \bowtie_{c.custid=i.customerid} \bowtie INVOICE\ i \bowtie$

$INVOICEDETAILS\ id))$

select c.custid, sum(id.rate) as rate from customer c inner join invoice i on c.custid=i.customerid natural join invoicedetails id group by c.custid, i.invno order by rate desc limit 1;

OR

select customerid from invoice where invno in (select invno from invoicedetails it natural join
items i where it.rate - i.averagepurchaseprice in (select max(it.rate - i.averagepurchaseprice) from invoicedetails it natural join items i)) ;

7. Top selling item (in terms of numbers) for a given year.

$_{it.code,\ it.name}\mathcal{F}_{SUM(id.qty)\ \rightarrow\ qty}\big(\sigma_{ORDER\ BY\ qty\ desc\ LIMIT\ 1}\big(\sigma$

$_{i.invdate\ BETWEEN\ '2011-01-01'\ AND\ '2011-12-31'}(ITEMS\ it$

$\bowtie_{it.code=id.itemcode} \bowtie INVOICEDETAILS\ id \bowtie INVOICE$

$i))$

select it.code, it.name, sum(id.qty) as qty from items it inner join invoicedetails id on it.code = id.itemcode natural join invoice i where i.invdate between '2011-01-01' and '2011-12-31' group by it.code, it.name order by qty desc limit 1;

On da-acad database

8. Retrieve ID of faculties who took more than one courses in a semester (for all semester in the database).

$$_{instructorid}\mathcal{F}\ _{COUNT(courseno)}\left(\sigma_{\ semester='Autumn'\ OR\ semester='Summer'}\right.$$

$$\left._{OR\ semester='Winter'}\left(\sigma_{\ COUNT(DISTINCT\ courseno)>1}(OFFERS)\right)\right)$$

select instructorid from offers where semester='Autumn' or semester='Summer' or semester='Winter' group by instructorid having count(distinct courseno)>1;

<div align="center">OR</div>

select instructorid from offers group by instructorid having count(distinct courseno) > 1;

9. List total count for each instructor. List the faculty name even if course count is zero.

$$_{i.instructorid,\ i.instructorname}\mathcal{F}_{COUNT(o.courseno)}\ (INSTRUCTOR\ i \bowtie_{i.instructorid\ =\ o.instructorid}\ OFFERS)$$

select i.instructorid, i.instructorname,count(o.courseno) from instructor i left join offers o on (i.instructorid=o.instructorid) group by (i.instructorid);

10. Retrieve all students (StudentID, Name, TotalCreditTaken) for B.Tech. (CS) (progid='02') batch 2007 in Autumn'2008.

$$_{s.studentid,\ s.name}\mathcal{F}_{SUM(c.credit)}\ (\sigma_{s.progid='02'\ AND\ s.batch=2007\ AND\ r.semester='Autumn'\ AND\ r.acadyear=2008}(STUDENT\ s \bowtie REGISTERS\ r \bowtie COURSE\ c))$$

select s.studentid, s.name,sum(c.credit) from student s natural join registers r natural join course c where s.progid='02' and s.batch=2007 and r.semester='Autumn' and r.acadyear=2008 group by s.studentid;

11. Retrieve all students (Id and name) who got more than two F grades in Autumn'2008.

$_{\text{s.studentid, s.name}}\mathcal{F}_{\text{COUNT(r.grade)}} (\sigma_{\text{r.acadyear=2008 AND r.grade='F'}} (\sigma_{\text{COUNT(grade)>2}} (\text{STUDENT s} \bowtie \text{REGISTERS r})))$

select s.studentid, s.name from student s natural join registers r where r.acadyear='2008' and r.grade in('F') group by s.studentid, s.name having count(r.grade)>2;