LAB-4 SOLUTIONS

SQL

1a)

```
select name from items as i join (select itemcode, sum(qty) as sales from salesdetails group by itemcode) as r on (i.code = r.itemcode) where (stock - sales) < reorderlevel;
```

1b)

```
select invdate, sum(day_sales) as total_day_sales from

(select invdate, qty*price as day_sales from sales natural join salesdetails) as r1

group by invdate;
```

1c)

```
select year, name from (

select customerid , name, extract(year from invdate) as year, sum(qty*price) as sales from

(customer as c join sales as s on (c.custid = s.customerid)) as r1

natural join salesdetails

group by customerid, name, year

) as x natural join (

select year, max(sales) as sales from

(select customerid , name, extract(year from invdate) as year, sum(qty*price) as sales from

(customer as c join sales as s on (c.custid = s.customerid)) as r3

natural join salesdetails

group by customerid, name, year) as r4 group by year) as r5
```

1d)

```
select itemcode, sum(qty) as qty_sold from salesdetails

group by itemcode

order by qty_sold desc

offset 1

limit 1
```

1e)

```
SELECT customerid, "year" FROM ( SELECT customerid, sum(profit), "year"
FROM ( SELECT

customerid, qty*(saleprice-averagepurchaseprice ) AS profit, EXTRACT ( YEAR
FROM invdate )

as "year" FROM (SELECT * FROM sales NATURAL JOIN salesdetails) AS r1 JOIN
items ON (

itemcode = code ) ) AS r3 GROUP BY customerid, "year" ) AS r4 WHERE
(sum,"year") in (SELECT

max(sum), "year" FROM ( SELECT customerid, sum(profit), "year" FROM (
SELECT customerid,

qty*(saleprice-averagepurchaseprice ) AS profit, EXTRACT ( YEAR FROM
invdate ) as "year"

FROM (SELECT * FROM sales NATURAL JOIN salesdetails) AS r1 JOIN items ON (
itemcode =

code ) ) AS r3 GROUP BY customerid, "year") AS r5 GROUP BY "year");
```

1f)

```
select distinct itemcode, yr, qty from customer natural join (select itemcode, extract(year from invdate) as yr, sum(qty) as qty from sales natural join salesdetails group by itemcode, yr ) as foo where qty in (select max(qty) from customer natural join (select itemcode, extract(year from invdate) as yr, sum(qty) as qty from sales natural join salesdetails group by itemcode, yr ) as foo group by yr );
```

1g)

```
set search_path to sales;

select * from items left join (select itemcode, name from salesdetails natural join (select * from sales join customer on sales.customerid=customer.custid) as r) as s on items.code = s.itemcode;
```

2a)

```
select instructorname, acadyear, semester, count(*) as n_courses from

(select * from instructor natural join offers) as r1

group by 1,2,3

having count(*) > 1;
```

2b)

```
select c.courseno, coursename from course as c

left join (select distinct courseno, instructorid from offers) as r

on c.courseno = r.courseno

where instructorid is NULL;
```

2c)

```
select studentid, name, acadyear, semester ,sum(credit) from( select * from student natural join registers natural join course natural join program where batch = 2007 and progname like 'Btech(CS)' ) as r group by 1,2,3,4 having (sum(credit) < 10 or sum(credit) > 20);
```

2d)

```
select studentid, name from( select studentid, name from student natural join registers where grade = 'FF') as r group by 1,2 having count(*) > 2;
```

2e)

```
SELECT * FROM student AS s
WHERE studentid NOT IN (
SELECT studentid FROM (
SELECT studentid, courseno, acadyear, semester from
((select courseno, acadyear, semester FROM offers NATURAL JOIN instructor
WHERE instructorname = 'P M Jat' AND acadyear >= 2007 AND acadyear <=
2011) as co
CROSS JOIN (select distinct studentid from registers) as sr)
EXCEPT
(SELECT studentid, courseno, acadyear, semester FROM
registers WHERE acadyear >= 2007 AND acadyear <= 2011)
) as r
);
```

RELATIONAL ALGEBRA

1a)

$r \leftarrow {}_{itemcode}\mathcal{F}_{sum(qty)\ as\ sales}(salesdetails)$

$r1 \leftarrow \pi_{name}(\sigma_{(stock - sales) < reorderlevel}(items_{(i)} \bowtie_{i.code = r.itemcode} r))$

1b)

$r1 \leftarrow \pi_{invdate,\ (qty*price)\ as\ day\_sales}(sales * salesdetails)$

$r2 \leftarrow {}_{invdate}\mathcal{F}_{sum(day\_sales)\ as\ total\_day\_sales}(r1)$

1c)

$r1 \leftarrow \pi_{customerid,\ name}({}_{extract(year\ from\ invdate)\ as\ year,\ sum(qty*price)\ as\ sales}\mathcal{F}(customer_{(c)} \bowtie_{c.custid = s.customerid} sales_{(s)})$

$X \leftarrow \pi_{year,\ name}(r1 * {}_{customerid,\ name,\ year}\mathcal{F}(salesdetails))$

$r3 \leftarrow r1$

$r4 \leftarrow \pi_{year}({}_{max(sales)}\mathcal{F}(r3 * {}_{customerid,\ name,\ year}\mathcal{F}(salesdetails)))$

$r5 \leftarrow {}_{year}\mathcal{F}(X * r4)$

1d)

$r1 \leftarrow {}_{itemcode}\mathcal{F}_{sum(qty)\ as\ qty\_sold}(salesdetails)$

1g)

$r \leftarrow \pi_{*}(sales_{(s)} \bowtie_{s.customerid = c.custid} customer_{(c)})$

$s \leftarrow \pi_{itemcode,\ name}(salesdetails) * r$

$r1 \leftarrow \pi_{*}(items_{(i)} \rtimes_{i.code = s.itemcode} s)$

2a)

$r1 \leftarrow \pi_{*}$ (instructor * offers)

$r2 \leftarrow \pi_{\text{instructorname, acadyear, semester}} ({}_{1, 2, 3}F_{\text{count(*)}} (\sigma_{\text{count(*)} > 1} (r1)))$

2b)

$r \leftarrow \pi_{\text{courseno, instructorid}}$ (offers)

$r1 \leftarrow \pi_{\text{c.courseno, c.coursename}} (\sigma_{\text{instructorid = NULL}} (\text{course}_{(c)} \bowtie_{\text{c.courseno = r.courseno}} r))$

2c)

$r \leftarrow \pi_{*}(\sigma_{\text{batch = 2007 AND program = 'Btech(CS)'}}$ (student * registers * course * program))

$r2 \leftarrow {}_{\text{studentid, name, acadyear, semester}}F_{\text{sum(credit)}} (\sigma_{\text{sum(credit) < 10 OR sum(credit) > 10}} (r))$

2d)

$r \leftarrow \pi_{\text{studentid, name}} (\sigma_{\text{grade = 'FF'}}$ (student * registers))

$r1 \leftarrow {}_{1, 2}F (\sigma_{\text{count(*) > 2}} (r))$

BY-
SHUBHAM ANNADATE (201501020)
NIRBHAY RAM (201501068)
ARNAV SHARMA (201501236)