COMP518 Assignment 3

William Cawley Gelling

201077658

I have included an easier to read solution to question one here with the txt file being used to copy and pasted into SQL.

1. Create Tables for the schema
   Enter the MySQL code for these tables.

   Schemas are :
   Employee(<u>eid,</u> ename, age)

   ```
   create table Employee (
   eid varchar(5) not null,
   ename varchar(30) not null,
   age smallint not null,
   primary key (eid));
   ```

   assumptions
   - Assuming employee eid fits into 5 Varchar in the form EXXXX where X represents any number.
   - Assume that people's names fit into 30 characters.
   - Chosen age to be small int as the age, could have date of birth column and then calculate age from it.
   - Assumed that non of these can be null when entered

   Department(<u>did,</u> dname, dtype, address)

   ```
   create table Department  (
   did varchar(5) not null,
   dname varchar(30) not null,
   dtype varchar(15) not null,
   address varchar(100) not null,
   primary key (did));
   ```

   assumptions
   - Did in the form DXXXX where x represents any number
   - Department name fits into 30 characters
   - Department type fits into 15 characters
   - Address fits into 100 characters
   - That non of these can be null

   WorksIn(<u>eid, did</u>, since)

```
create table WorksIn (
eid varchar(5) not null,
did varchar(5) not null,
since date not null,
primary key (eid, did),
foreign key (eid) references Employee (eid) on delete cascade on update cascade,
foreign key (did) references Department (did) on delete cascade on update cascade);
```

assumptions
- Since is a date
- Decided that on delete they cascade through.
- On update they also cascade

Product(pid, pname, ptype, pcolor)

```
create table Product (
pid varchar(5) not null,
pname varchar(30) not null,
ptype varchar(20) not null,
pcolor varchar(20) not null,
primary key (pid));
```

assumptions
- pid in the form PXXXX where x represents any number
- product name fits into 30 characters
- ptype type fits into 20 characters
- pcolor into 20 characters
- That non of these can be null

Sells(did, pid, quantity)

```
create table Sells (
did varchar(5) not null,
pid varchar(5) not null,
quantity integer not null default 0,
primary key (did, pid),
foreign key (did) references Department(did) on delete cascade on update cascade,
foreign key (pid) references Product(pid) on delete cascade on update cascade);
```

assumptions
- Decided that on delete they cascade through.
- - quantity is an integer and its default is 0
- On update they cascade

2. (a) Find the names of departments which sell blue products?

```
select distinct D.dname
```

```
 from Department D, Product P, Sells S
where P.pid = S.pid and S.did = D.did and pcolor = 'blue';
```

(b) find the names of departments which sell products of type tool and product of type toy?

```
select D.dname
from Department D
where D.did in
        (select S.did from Product P, Sells S where P.pid = S.pid and P.ptype = 'tool' and S.did
        in (select S.did from Product P, Sells S where P.pid = S.pid and P.ptype = 'toy'));
```

c) find the names of departments which sell blue products and do not have any employees over the age of 40

```
select D.dname
from Department D
where D.did in
        ( select S.did from Sells S, Product P where P.pid = S.pid and P.pcolor = 'blue'  and
        S.did not in
                ( select W.did from WorksIn W, Employee E where W.eid = E.eid and E.age >
                40));
```

d) For each department report the department-id and the age of the oldest employee working in it.

For this we are making the assumption that if the department has no one working in it then there is no age to state. Therefore it is not included

```
select W.did, max(E.age) as "Oldest Employee's Age"
from WorksIn W, Employee E
where W.eid = E.eid group by W.did;
```

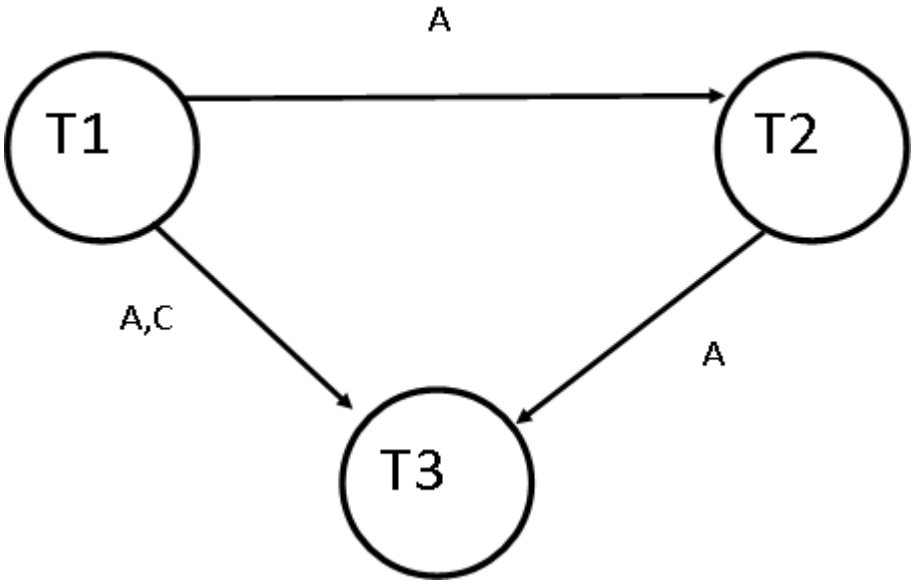
e) Find the names of employees who are older than at least one employee working in department `Central'.

```
select E.ename from Employee E
where E.age > (select min(E.age)
        from Employee E, WorksIn W, Department D
        where D.Dname = 'Central' and D.did = W.did and W.eid = E.eid);
```

f) Find the names of employees working in departments which have sold at least 5 types of products.

```
select E.ename
from Employee E, WorksIn W
where E.eid = W.eid and W.did in (select S.did
                from Sells S group by S.did having count(*) > 4);
```

2.1.i)



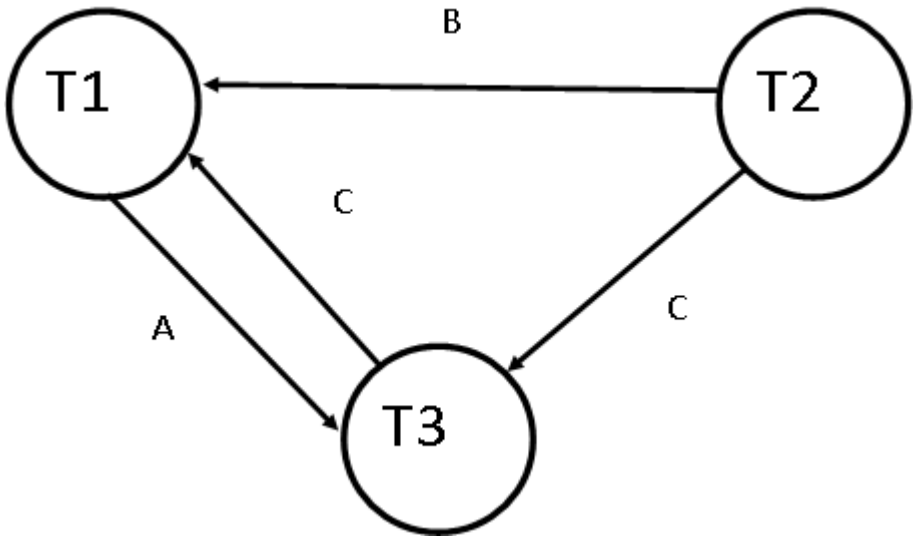
ii) no cycle, so it is conflict serializable, a corresponding serial schedule would would be T1,T2,T3

R1(A),R1(B),W1(A),R1(C),W1(c), R2(A),W2(A),R3(C),R3(B),W3(A)

iii) yes this schedule can occur using 2PL as follows:

read_lock(T1, A), read(T1,A), read_lock(T1, B), read(T1,B), write_lock(T1,A),write(T1,A), read_lock(T1,C), write_lock(T1,C), unlock(T1,A), read_lock(T2,A), read(T2,A), read(T1,C), write(T1, C), unlock(T1,B), unlock(T1,C), read_lock(T3,C), read(T3,C), write_lock(T2, A), write(T2,A), unlock(T2,A), read_lock(T3,B), read(T3, B), write_lock(T3,A), write(T3,A), unlock(T3,C), unlock(T3,B), unlock(T3,A)

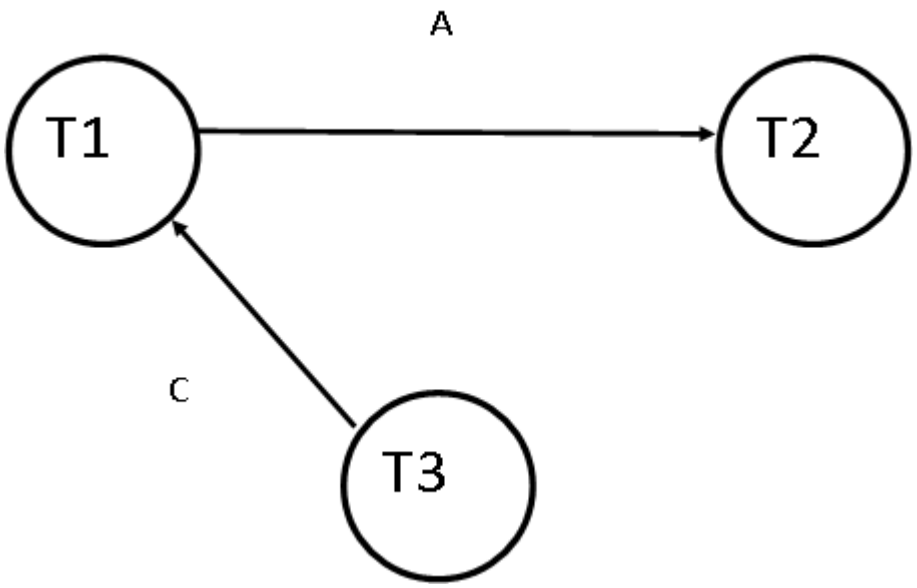
2.2.i)



ii) Cycle so not conflict serializable

iii) Cycle so not conflict serializable and therefore cannot use 2pl

2.3.i)



ii) no cycle, so it is conflict serializable, a corresponding serial schedule would would be T3,T1,T2

W3(C), R1(A), R1(B),W1(A),W1(C),R2(A),W2(A)

iii) yes this schedule can occur using 2PL as follows:

read_lock(T1,A),read(T1,A), read_lock(T1,B),read(T1,B),), write_lock(T1,A),write(T1,A),read_lock(T2,A), read(T2,A), write_lock(T3,C),write(T3,C), unlock(T3,C), write_lock(T1,C),write(T1,C), unlock(T1,A), unlock(T1,B), unlock(T1,C), write_lock(T2,A), write(T2, A), unlock(T2,A)

3.1)

Time	A	B	Product
0	3	5	n/a
1	3	5	n/a
2	3	5	n/a
3	3	5	1
4	3	5	1
5	1	5	1
6	1	5	3
7	1	5	3
8	1	5	3
9	1	5	3
10	1	6	3
11	1	6	3
12	2	6	3
13	2	6	3
14	2	6	3
15	2	6	21
16	2	6	21

ii) a = 0, b = 6 product = 7

iii)a = 0 , b = 6, product = 18