TP N° 5-Les Vues

Exercice 1

1.Tables Creation

Client

```
create table clientTp5 (
numcl number,
nom varchar2(50),
prenom varchar2(50),
genere varchar2(15),
adresse varchar2(50),
tel varchar2(50),
constraint pk_clientTP5 primary key (numcl),
constraint chk_clienTP5_genere check (genere in ('homme','femme'))
);
```

Agence

```
create table agenceTP5 (
idagence number,
adresse varchar2(50),
responsable varchar2(50),
constraint pk_agenceTP5 primary key (idagence)
);
```

Voiture

```
create table voitureTP5 (
matricule number,
modele varchar2(50),
marque varchar2(50),
couleur varchar2(50),
description varchar2(50),
idagence number,
constraint pk_voitureTP5 primary key (matricule),
constraint fk_voitureTP5_agenceTP5 foreign key (idagence) references agencetp5(idagence)
);
```

Louer

```
create table louerTP5 (
idagence number,
numcl number,
matricule number,
datedebut date,
datefin date,
prix number,
typelocation varchar2(50),
constraint pk_louerTP5 primary key (idagence,numcl,matricule,datedebut),
constraint fk_louerTP5_agenceTP5 foreign key (idagence) references agenceTP5(idagence),
constraint fk_louerTP5_clientTP5 foreign key (numcl) references clientTP5(numcl),
constraint fk_louerTP5_voitureTP5 foreign key (matricule) references voitureTP5(matricule),
constraint chk_louerTP5_date check (datefin>datedebut),
constraint chk_louerTP5_type check (typelocation in ('longue duree' ,'courte duree'))
);
```

Insert

i can't put the insert queries in this pdf because they have a lot of lines , you can find all the insert sql code inside the SQL/EX1/insert/ folder

2.Client That Never Ranted For Longue Period

```
--method with nested select
select c.nom , c.prenom from clienttp5 c where c.numcl not in (
 select l.numcl from louertp5 l where l.typelocation='longue duree'
);

--method with left join
select  c.nom , c.prenom from clienttp5 c LEFT JOIN louertp5 l on c.numcl = l.numcl and
l.typelocation='longue duree'
where l.numcl is  null;
```

Nested Select Method

Inner Query

```
select l.numcl from louertp5 l where l.typelocation='longue duree'
```

This inner query fetches set of numcl that at least ranted a car for a 'longue duree' once

Outer Query

```
select c.nom , c.prenom from clienttp5 c where c.numcl not in ( inner query );
```

This outer query select first name and last name of the client that didn't do 'longue duree' at least once → meaning client that never ranted for 'longue duree'

Joing Method

Join

```
    select  c.nom , c.prenom from clienttp5 c LEFT JOIN louertp5 l
```

We are doing left join between client and louer , so all client from client table will be fetched and if the is no match with louer table it will return null

Join Condition

```
    on c.numcl = l.numcl and l.typelocation='longue duree'
```

This join condition macthes for client that exist on louer table and ranted for 'longue duree' , so it would return null if client doesn't exist in louer table or didn't ranted for 'longue duree' (at least once instance)

Where Clause

```
    where l.numcl is null;
```

Null Client mean client that didn't match → don't exist in louer table or never ranted for 'longue duree'

3.Algeria Agency That Offer 'BMW' Or 'Volkswagen' Cars Only

```
 --method nested select
select a.idagence from agencetp5 a where a.adresse='Alger'
and (a.idagence not in (select v.idagence from voituretp5 v where v.marque not in ('BMW','Volkswagen')) --
    agency do both only
or a.idagence not in (select v.idagence from voituretp5 v where v.marque not in ('BMW')) --agency that does
    BMW only
or a.idagence not in (select v.idagence from voituretp5 v where v.marque not in ('Volkswagen')) --agency that
    does Volkswagen only
);

 --method join
select distinct a.idagence from
agencetp5 a inner join voituretp5 v on a.idagence = v.idagence and a.adresse='Alger'
left join voituretp5 v1 on v1.idagence = a.idagence and
(v1.marque not in ('BMW','Volkswagen'))
where v1.idagence is null ;
```

Nested Select Method

First Inner Query

```
select v.idagence from voituretp5 v where v.marque not in ('BMW','Volkswagen')
```

We are fetching idagence from voiture table that doesn't offer cars 'BMW' or 'Volkswagen' at least once

Second Inner Query

```
a.idagence not in (select v.idagence from voituretp5 v where v.marque not in ('BMW'))
```

We are fetching idagence from voiture table that doesn't offer cars 'BMW' at least once

Third Inner Query

```
a.idagence not in (select v.idagence from voituretp5 v where v.marque not in ('Volkswagen')
```

We are fetching idagence from voiture table that doesn't offer cars 'Volkswagen' at least once

Outer Query

```
select a.idagence from agencetp5 a where a.adresse='Alger'and
(a.idagence not in (first inner query)
or a.idagence not in (second inner query)
or a.idagence(third inner query));
```

We are fetching idagence from agence table of Algeria that !(doesn't offer cars 'BMW' or 'Volkswagen' at least once) OR !(doesn't offer cars 'BMW' at least once) OR !(doesn't offer cars 'Volkswagen' at least once) → agency that offers 'BMW' and 'Volkswagen' Only OR agency that offers 'BMW' OR agency that offers 'Volkswagen' only → agency that offers 'BMW' and 'Volkswagen'

Joins Method

Inner Join

```
select distinct a.idagence from  agencetp5 a inner join voituretp5 v
on a.idagence = v.idagence and a.adresse='Alger'
```

This inner join fetches distinct idagence that exist on voiture table and are located in 'Algeria' , we used distinct because this inner join can match the same agency multiple time (agency that offers >1)

Left Join

```
inner_Join_Output left join voituretp5 v1
on v1.idagence = a.idagence and v1.marque not in ('BMW','Volkswagen')
```

This left join matches the output of first inner join with voiture table , condition join : if agency exist on voiture table and if it doesn't offer 'BMW' or 'Volkswagen' cars at least once , it returns null if agency doesn't exist in voiture (but that is impossible due to first inner join) or the agency always offers 'BMW' or 'Volkswagen' cars

Where Clause

```
where v1.idagence is null;
```

This makes sure only agency that aren't matched from the left join that are returned → agency that offers only 'BMW' or 'Volkswagen' cars

4.Most Loyal Client

```
create view v1TP5EX1(idagence,numcl,nblocation) as select l.idagence,l.numcl ,count(l.numcl) nblocation
from louertp5 l group by l.idagence,l.numcl
having count(l.numcl) = (
select max(count(l2.numcl)) from louertp5 l2 where l2.idagence = l.idagence group by l2.numcl
);
```

Nested Select Method

Inner Query

```
select max(count(l2.numcl)) from louertp5 l2 where l2.idagence = l.idagence group by l2.numcl
```

It fetches the max rants for each agency

Outer Query

```
create view v1TP5EX1(idagence,numcl,nblocation) as select l.idagence,l.numcl ,count(l.numcl) nblocation
from louertp5 l group by l.idagence,l.numcl
having count(l.numcl) = inner query
```

Creates external view v1 with idagence,numcl,nblocaton from tables louer and groups by idagence,numcl and we filter it with having by making sure each client from same agency must have max ranted count

Can We Insert On V1?

We can't insert on the table because the select query has an aggregation function : count and max

5.NBL

Add NBL column

```
alter table clienttp5 add nbl number default 0;
```

Fill NBL For Already Existing Client

```
create or replace procedure fillNBL

is

cursor cr is
select numcl from clienttp5;

cl number;

begin
open cr;
fetch cr into cl;

while cr%found loop

update clienttp5 c set c.nbl = ( select count(l.numcl) from louertp5 l where l.numcl=cl ) where c.numcl = cl;
fetch cr into cl;

end loop;

close cr;

exception
when others then
dbms_output.put_line(sqlcode||sqlerrm);
end;
/
```

FillNBL procedure

we are using cursor to loop through each client in client table and for each client we fetch their number of rants and update the value of nbl with it

After Insert Trigger

```
create or replace trigger insertLouerTP5
after insert on louerTP5
for each row

begin

update clienttp5 c set c.nbl = c.nbl+1 where c.numcl = :new.numcl;

end;
/
```

Insert Trigger

We are creating row level trigger (for each row) , because we need acces to :new to be able to increment nbl for each client that was inserted in louer table

Before Insert Trigger

```
create or replace trigger deleteLouerTP5
after delete on louerTP5
for each row

begin

update clienttp5 c set c.nbl = c.nbl-1 where c.numcl = :old.numcl;

end;
/
```

Delete Trigger

We are creating row level trigger (for each row) , because we need acces to :new to be able to decrement nbl for each client that was deleted from louer table

Exercice 2

1.Tables Creation

Chef

```
create table chefTP5 (
numc number,
nomc varchar2(50),
age integer,
pays varchar2(50),
specialiteCulinaire varchar2(50),
etoilesc integer,
constraint pk_chefTP5 primary key (numc),
constraint chk_chefTP5_age check (age>=18),
constraint chk_chefTP5_etoilesc check (etoilesc in (1,2,3,4,5))
);
```

Restaurant

```
create table restaurantTP5 (
numr number,
nomr varchar2(50),
adresse varchar2(100),
etoilesr integer,
constraint pk_restaurantTP5 primary key (numr),
constraint chk_restaurantTP5_etoilesr check (etoilesr in (1,2,3,4,5))
);
```

Service

```
create table serviceTP5(
numc number,
numr number,
dateService date,
constraint pk_serviceTP5 primary key (numc,numr,dateservice),
constraint fk_serviceTP5_chefTP5 foreign key (numc) references cheftp5(numc),
constraint fk_serviceTP5_restaurantTP5 foreign key (numr) references restauranttp5(numr)
);
```

Insert

All insert queries are inside SQL/EX2/insert folder

Chefs That Worked In All Oran Restaurant(And Maybe Others Too)

```
create view vcheftp5 as select c.nomc from cheftp5 c where
(select count(s.numr) from servicetp5 s where s.numc = c.numc and
s.numr in
(select r.numr from restauranttp5 r where r.adresse='Oran')) =
(select count(r.numr) from restauranttp5 r where r.adresse='Oran');
```

Nested Select Method

First Inner Query

```
select count(s.numr) from servicetp5 s where s.numc = c.numc and
s.numr in  (select r.numr from restauranttp5 r where r.adresse='Oran')
```

Calculate number of 'Oran' Restaurant for each chef (s.numc = c.numc) has worked in by using count numrestaurant and filtering with where to count oran restaurant only

Second Inner Query

```
select count(r.numr) from restauranttp5 r where r.adresse='Oran'
```

Calculate number of 'Oran' restaurant that exist in restaurant table

Outer Query

```
create view vcheftp5 select c.nomc from cheftp5 c where
(number of 'Oran' restaurant a chef worked in) =
(Total number of 'Oran' restaurant);
```

We are creating a view that fetches chef name that worked in all 'Oran' Restaurant

Can We Insert On VChefTP5

No we can't insert because the select has aggregation function count

Chefs That Worked In Maximums Of Differents Restaurants

```
create view vmaxcheftp5 as select c.nomc ,
(select count(distinct s.numr) from servicetp5 s where s.numc= c.numc) max from cheftp5 c
where(select count(distinct s.numr) from servicetp5 s where s.numc= c.numc)=
(select max(count(distinct s.numr)) from servicetp5 s group by s.numc);
```

Nested Select

First Inner Query

```
select count(distinct s.numr) from servicetp5 s where s.numc= c.numc
```

Calculate the number of distinct restaurant a chef (s.numc=c.numc) has worked in

Second Inner Query

```
select max(count(distinct s.numr)) from servicetp5 s group by s.numc
```

Calculate the max number of distinct restaurant chefs have worked in , **note** the group by is curcial so we calculate number of distinct restaurant for each chef then pass it to max

Outer Query

```
create view vmaxcheftp5 as select c.nomc ,
( number of distinct restaurants a chef has worked in ) max from cheftp5 c
where ( number of distinct restaurants a chef has worked in ) =
(max number of distint restaurants chefs have worked in );
```

Creating a view that fetches name of chef that worked in maximum distinct restraurants , **note** we have to redo the query to be able to print the number because where can't acess aliases because they are executed before select

Can We Insert On VMaxChefTP5

No we can't insert because the select has aggregation function count , max

Exercice 3

Exercice 4