

TP N° 4-Les Triggers

1.Trigger After Employee Inserted

```
1 create or replace trigger tp4q1
2 after insert on employe
3
4 begin
5
6 dbms_output.put_line('Un Nouveau Employe Est Ajoute');
7
8 end;
9 /
```

TP4Q1 Trigger

We created a table level trigger (no need for :new , :old) to print 'Un Nouveau Employe Est Ajoute' after an employee is inserted

2.Trigger After Product Inserted

```
1 create or replace trigger tp4q2
2
3 after insert on produit
4 for each row
5
6 declare
7 nom categorie.nomcateg%type;
8
9 begin
10
11 select cat.nomcateg into nom from categorie cat where cat.codecateg = :new.codecateg;
12
13 dbms_output.put_line('Un Produit Est Ajaoute A La Categorie '||nom);
14
15 end;
16 /
```

TP4Q2 Trigger

We created a row level trigger (need :new to fetch category of product) to print ('Un Produit Est Ajaoute A La Categorie '||nom) after a product is inserted

3.Trigger Before codeclient is updated on commande

```
1 create or replace trigger tp4q3
2 before update of codeclient on commande
3 for each row
4
5 declare
6 messageError varchar2(100);
7 exist number;
8 exc exception;
9
10 begin
11 messageError:='Error Can''t Update CodeClient In Commnade Table Because '||:new.codeclient||' Doesn''t exist
12     in Client Table';
13
14 select count(codeclient) into exist from client where codeclient = :new.codeclient;
15
16 exception
17 when NO_DATA_FOUND then
18 dbms_output.put_line(messageError);
19 when others then
20 dbms_output.put_line(sqlcode||sqlerrm);
21 end;
22 /
```

TP4Q2 Trigger

We created a row level trigger to get acces to :new.codeclient and be able to verify whether it exist in client table , if the exception no_data_found is raised it means the new client code doesn't exist and we print the error message

4.Chiffre Affaire Of Product

4.a.Add Chiffre_Affaire Column To Product Table

```
1 alter table produit add chiffre_affaire number default 0;
```

4.b.Update Chiffre_Affaire Automatically

After Insert On DetailCommande

```
1 create or replace trigger insertProdDC
2 after insert on detailcommande
3 for each row
4
5 declare
6 newca number;
7
8 begin
9
10 newca:=:new.qte*:new.prixunit-:new.remise;
11
12 update produit p set p.chiffre_affaire = p.chiffre_affaire + newca
13 where p.refprod = :new.refprod;
14
15 end;
16 /
```

insertProdDC

We created a row level trigger after insert on detailcommande , we need to increment chiffre affaire of the new inserted product we need to first calculate how much the product made

```
1 newca:=:new.qte*:new.prixunit-:new.remise;
```

Then we increment chiffre affaire of the newly inserted product with the calculated value

```
1 update produit p set p.chiffre_affaire = p.chiffre_affaire + newca
2 where p.refprod = :new.refprod;
```

After Delete On DetailCommande

```
1 create or replace trigger deleteProdDC
2 after delete on detailcommande
3 for each row
4
5 declare
6 oldca number;
7
8 begin
9
10 oldca:=:old.qte*:old.prixunit-:old.remise;
11
12 update produit p set p.chiffre_affaire = p.chiffre_affaire - oldca
13 where p.refprod = :old.refprod;
14
15 end;
16 /
```

deleteProdDC

We created a row level trigger after delete on detailcommande , we need to decrement chiffré affaire of the deleted product we need to first calculate how much the deleted product made

```
1 oldca:=:old.qte*:old.prixunit-:old.remise;
```

Then we decrement chiffré affaire of the deleted product with the calculated value

```
1 update produit p set p.chiffré_affaire = p.chiffré_affaire - oldca  
2 where p.refprod = :new.refprod;
```

Note

We can't use a select query to fetch the newca/oldca from detailcommand table

Because it would result in a mutated table error since trigger would read from a table that is being modified

After Update On DetailCommande

```
1  create or replace trigger modifProdDC
2  after update on detailcommande
3  for each row
4
5  declare
6  oldca number;
7  newca number;
8
9  begin
10
11  oldca:= :old.prixunit*:old.qte-:old.remise;
12  newca:= :new.prixunit*:new.qte-:new.remise;
13
14
15  if (:old.refprod != :new.refprod) then
16
17  update produit p set p.chiffre_affaire = p.chiffre_affaire -oldca
18  where p.refprod = :old.refprod;
19
20  update produit p set p.chiffre_affaire = p.chiffre_affaire + newca
21  where p.refprod = :new.refprod;
22
23  else
24  update produit p set p.chiffre_affaire = p.chiffre_affaire - oldca + newca
25  where p.refprod = :old.refprod;
26  end if;
27
28
29  end;
30  /
```

modifProdDC

We created a row level trigger before update on detailCommande , we can divide it into 2 cases :

Product Got Replaced

If the product got replaced with another one (:old.refprod!=:new.refprod) then it means we need to subtract from the deleted product (:old.refprod)

```
1  update produit p set p.chiffre_affaire = p.chiffre_affaire -oldca
2  where p.refprod = :old.refprod;
```

and increment the new product (:new.refprod)

```
1  update produit p set p.chiffre_affaire = p.chiffre_affaire + newca
2  where p.refprod = :new.refprod;
```

Same Product

Now if the product is the same and only other column changed like remise , price ...etc we remove from chiffre_affaire the old money made from that row for the product and increment it with the new one

```
1  update produit p set p.chiffre_affaire = p.chiffre_affaire - oldca + newca
2  where p.refprod = :old.refprod;
```