



Exercices

1. Ecrire un programme qui calcule le montant de la commande numéro 10 et affiche un message 'Commande Normale' ou 'Commande Spéciale' selon que le montant est inférieur ou supérieur à 100000 DH
2. Ecrire un programme qui supprime l'article numéro 8 de la commande numéro 5 et met à jour le stock. Si après la suppression de cet article, la commande numéro 5 n'a plus d'articles associés, la supprimer.
3. Ecrire un programme qui affiche la liste des commandes et indique pour chaque commande dans une colonne Type s'il s'agit d'une commande normale (montant ≤ 100000 DH) ou d'une commande spéciale (montant > 100000 DH)
4. A supposer que toutes les commandes ont des montants différents, écrire un programme qui stocke dans une nouvelle table temporaire les 5 meilleures commandes (ayant le montant le plus élevé) classées par montant décroissant (la table à créer aura la structure suivante : NumCom, DatCom, MontantCom)
5. Ecrire un programme qui :
 - ✓ Recherche le numéro de commande le plus élevé dans la table commande et l'incrémente de 1
 - ✓ Enregistre une commande avec ce numéro
 - ✓ Pour chaque article dont la quantité en stock est inférieure ou égale au seuil minimum enregistre une ligne de commande avec le numéro calculé et une quantité commandée égale au triple du seuil minimum

```

delimiter $$
CREATE procedure q1(cc int)
begin
declare total int;

select sum(a.PU * lc.QteCommandee) into total
from article a, lignecmd lc
where a.codeArt=lc.codeArt and lc.codeCmd=cc;

if total <1000 THEN

```

```

        select concat('codeCmd : ', cc, ' -Total : ', total, ' - remarque : Commande normal');
    ELSE
        select concat('codeCmd : ', cc, ' -Total : ', total, ' - remarque : Commande Spécial');
    end if;
end$$
call q1(10) ;$$

```

Dans le processus de l'achat
 Lors de l'achat de 100 unité de l'article 8, on **incrémente** le stock par 100
 Lorsqu'on annule l'achat on **décrémente** le stock

```

delimiter $$
CREATE procedure q2_AnnulerAchat(cc int, ca int)
begin
    DECLARE qt int;
    declare nbr int;
    select lc.QteCommandee into qt from lignecmd lc
    where lc.codeCmd=cc and lc.codeArt=ca;

    update article set QteEnStock =QteEnStock -qt;
    DELETE from lignecmd where codeCmd=cc and codeArt=ca;
    select count(*) into nbr from lignecmd where codeCmd=cc;
    if nbr=0 THEN
        delete from commande where codeCmd=cc;
        select concat('La commande : ', cc, ' ne contient plus d'article,elle sera supprimée') as
        'remarque1';
    ELSE
        select concat('La commande : ', cc, ' contient encore : ',nbr, ' articles') as 'remarque2';
    end if;
end$$

call q2_AnnulerAchat(89,1 )

```

```

delimiter $$
drop procedure q2_AnnulerAchat;$$
CREATE procedure q2_AnnulerAchat(cc int, ca int)
begin
    DECLARE qt int;
    declare nbr int;
    declare nbrCmd int;
    select count(*) into nbrCmd from commande where codeCmd=cc;
    if nbrCmd>0 then
    begin
        select lc.QteCommandee into qt from lignecmd lc
        where lc.codeCmd=cc and lc.codeArt=ca;

        update article set QteEnStock =QteEnStock -qt where codeArt=ca;

        DELETE from lignecmd where codeCmd=cc and codeArt=ca;
        select count(*) into nbr from lignecmd where codeCmd=cc;
        if nbr=0 THEN
            delete from commande where codeCmd=cc;
            select concat('La commande : ', cc, ' ne contient plus d'article,elle sera supprimée') as
            'remarque1';

```

```

ELSE
    select concat('La commande : ',cc,' contient encore : ',nbr,' articles') as 'remarque2';
end if;
end;
ELSE
    SELECT concat('la commnde : ', cc, ' n''existe pas') as 'remarque';

end if;
end$$

```

```

delimiter $$
CREATE PROCEDURE Q3()
begin
select c.dateCmd, c.codeCmd,sum( a.PU*Ic.QteCommandee) as 'Montant',
CASE
    when sum( a.PU*Ic.QteCommandee)<10000 then 'Commande normale'
    ELSE
        'Commande spéciale'
end as 'remarque'
from commande c,lignecmd lc, article a
where a.codeArt=lc.codeArt and c.codeCmd=lc.codeCmd
group by c.dateCmd, c.codeCmd
order by sum( a.PU*Ic.QteCommandee);
end$$

```

```

create temporary TABLE MaxCmd as
select c.dateCmd, c.codeCmd,sum( a.PU*Ic.QteCommandee) as 'Montant'
from commande c,lignecmd lc, article a
where a.codeArt=lc.codeArt and c.codeCmd=lc.codeCmd
group by c.dateCmd, c.codeCmd
order by sum( a.PU*Ic.QteCommandee) desc
limit 5;

```

```

create temporary TABLE test(codetest int , nomtest varchar(20));
insert into test values (10,'azerty'), (20,'www'), (30,'mlkjh');
select * from test;

```

```

Delimiter $$
Create procedure Q5()
Begin
declare CodeMax int;
declare dt date;
select CURRENT_DATE into dt;
select max(codeCmd) into codeMax from commande;
set codeMax=codemax+1;
insert into commande values(codeMax, dt);
insert into lignecmd select codeMax ,a.codeArt ,a.SeuilMin*3 from article a
where a.QteEnStock <a.SeuilMin;
End$$

```