

SGBD Rendu Contrôle Certification Professionnelle

Créer la base de données :

```
if exists ( select * from sys.databases where name='CertificationDb')
drop database CertificationDb;
create database CertificationDb;
use CertificationDb;
```

NB: Pour obtenir un certificat Le candidat doit avoir une note supérieure ou égale à 700.

1. Créer les tables de la base de données sachant que la première colonne de chaque table est une clé primaire, les champs précédés par # sont des clés étrangères. (2 pts)

Les conditions suivantes doivent être imposées :

- Le « codeCandidat » doit s'incrémenter automatiquement. (0.5 pt)
- Le « codeCertif » doit avoir le format suivant : 2 chiffres suivis d'un tiret suivi de 3 chiffres. (0.5 pt)
- La « note » doit être entre 0 et 1000. (0.5 pt)

```
create table Candidat (
    CodeCandidat int primary key identity(111,1),
    Nom varchar(30),
    Prenom varchar(30),
    DateNaissance date,
    CodePostal int,
    Ville varchar(30)
);
create table Certificat (
    CodeCertif varchar(6) primary key check (CodeCertif like '__-__-__'),
    Intitule varchar(300),
    NombreQuestions int,
    Prix money
);
create table Passer (
    Numero int primary key,
    CodeCandidat int foreign key references Candidat(CodeCandidat)
    on delete set null on update cascade,
    CodeCertif varchar(6) foreign key references Certificat(CodeCertif)
    on delete set null on update cascade,
    DatePassation date,
    Note float check (Note between 0 and 1000)
);
```

2. Introduire deux lignes dans chaque table. (0.5 pt)

```
insert into Candidat values
('BEKKALI', 'Jamil', '1986-03-11', 14525, 'IFRANE'),
('MOUAFI', 'Ihssane', '1974-05-23', 11000, 'SALE'),
('MOURTADA', 'Mohamed', '1990-05-01', 12500, 'SIDI KACEM'),
('NEJJAR', 'Idriss', '1987-03-25', 10200, 'MARRAKECH'),
('BAROUDI', 'Firdaouss', '1990-03-02', 10500, 'KENITRA');
insert into Certificat values
('70-432', 'Microsoft SQL Server 2008, Implementation and Maintenance', 55, 90),
('70-461', 'Querying Microsoft SQL Server 2012', 45, 85),
```

```
('70-464', 'Developing Microsoft SQL Server 2012 Databases', 50, 100),  
( '70-483', 'Programming in C#', 45, 90),  
( '98-375', 'HTML5 App Development Fundamentals', 50, 95);  
insert into Passer values  
(10101, 112, '70-432', '2015-05-03', 685),  
(10102, 114, '70-461', '2015-06-04', 750),  
(10103, 111, '70-432', '2015-05-03', 850),  
(10104, 114, '98-375', '2015-06-11', 575),  
(10105, 115, '70-464', '2015-06-06', 810),  
(10106, 112, '70-432', '2015-07-17', 700),  
(10107, 111, '70-432', '2015-07-17', 750);  
-- Affichages  
select * from Candidat;  
select * from Certificat;  
select * from Passer;
```

3. Afficher le nombre d'examens de certification passé pour chaque candidat. (1pt)

```
select CodeCandidat, count(*) as 'Nombre d'examens de certification passés'  
from Passer group by CodeCandidat;
```

4. Afficher le nombre de candidats qui ont obtenu au moins un certificat. (1 pt)

```
select count(distinct CodeCandidat) as 'Nombre de candidats qui ont obtenu au moins  
un certificat'  
from Passer where Note >= 700;
```

5. Afficher pour chaque certificat (Intitulé) la recette totale obtenue par la société (montant payé par tous les candidats) au mois de Mai 2015. (1 pt)

```
select Certificat.Intitule, sum(Certificat.Prix) as 'Recette totale (DH)'  
from Certificat inner join Passer on Certificat.CodeCertif = Passer.CodeCertif  
where month(Passer.DatePassation)=5 and year(Passer.DatePassation)=2015  
group by Certificat.Intitule;
```

6. Augmenter le prix de certificat de : (1 pt)

a. 10% si le nombre de questions est inférieur à 50

```
update Certificat set Prix += 0.1*Prix  
where NombreQuestions < 50;
```

b. 7% si le nombre de questions est entre 50 et 54

```
update Certificat set Prix += 0.07*Prix  
where NombreQuestions between 50 and 54;
```

c. 5% si le nombre de questions est supérieur à 54

```
update Certificat set Prix += 0.05*Prix  
where NombreQuestions > 54;
```

• Autre Méthode :

```
update Certificat set Prix += (  
    case  
        when NombreQuestions < 50 then 0.1*Prix  
        when NombreQuestions between 50 and 54 then 0.07*Prix  
        when NombreQuestions > 54 then 0.05*Prix  
    end
```

```
);
```

7. Supprimer les candidats qui n'ont jamais obtenu un certificat. (1 pt)

```
delete from Candidat where CodeCandidat not in  
(select distinct CodeCandidat from Passer where Note >= 700);
```

- Au cas où le comportement des clés étrangères lors de la suppression n'est pas définis :

```
alter table Passer  
drop constraint fk_CodeCandidat;  
alter table Passer  
add constraint fk_CodeCandidat foreign key (codeCandidat) references  
Candidat(codeCandidat) on delete set null on update cascade;  
delete from Candidat where CodeCandidat not in  
(select distinct CodeCandidat from Passer where Note >= 700);
```

8. Créer une procédure stockée qui affiche les candidats qui n'ont jamais passé un certificat. (1.5 pt)

```
create proc sp_candi_no_certi as  
select * from Candidat where CodeCandidat not in (select distinct CodeCandidat from  
Passer);  
-- Exécuter :  
exec sp_candi_no_certi;
```

9. Créer une procédure qui affiche le nombre de candidats qui ont réussi le certificat dont le code est passé en paramètre. (1.5 pt)

```
create proc sp_nbr_candi_certi  
@CodeCertif varchar(max)  
as  
    select count(distinct CodeCandidat) as 'Nombre de candidats qui ont réussi le  
certificat'  
    from Passer where Note >= 700 and CodeCertif = @CodeCertif;  
-- Exécuter :  
exec sp_nbr_candi_certi '70-432';
```

10. Créer une procédure stockée qui affiche les examens passés par un candidat entre deux dates. (1.5 pt)

```
create proc sp_nbr_exam_candi_dat  
@CodeCandidat int, @date_min date, @date_max date  
as  
    select * from Certificat where CodeCertif in (  
        select CodeCertif from Passer  
        where (DatePassation between @date_min and @date_max)  
        and CodeCandidat = @CodeCandidat  
    );  
-- Exécuter :  
exec sp_nbr_exam_candi_dat 114, '03-05-2015', '05-06-2015';
```

11. Modifier la procédure stockée de la question 10 de tel sort que la date de début doit être antérieure à la date de fin. (1 pt)

```
alter proc sp_nbr_exam_candi_dat  
@CodeCandidat int, @date_min date, @date_max date  
as
```

```
    if (@date_min > @date_max)
        print 'Attention! La première date doit être inférieure à la deuxième
!';
    else
        select * from Certificat where CodeCertif in (
            select CodeCertif from Passer
            where (DatePassation between @date_min and @date_max)
            and CodeCandidat = @CodeCandidat
        );
-- Exécuter :
exec sp_nbr_exam_candi_dat 111, '03-05-2016', '05-06-2015';
```

12. Supprimer la procédure de la question 9. (1 pt)

```
drop proc sp_nbr_candi_certi;
```