 **Direction Régionale Nord-Ouest I**

**OFPPT** -----------------------------------------------------------------------

**Examen de Fin de Module Régional \_ Session FEVRIER 2019**

**Module 08 : Base de Données**

**Filière : TDI Epreuve : Théorique**

**Niveau : TS Durée : 2 heures**

**Année  : 2ème A Barème : /40**

1. **Créer la base de données. (3 pts)**

createdatabasegestion\_Reservation

--

usegestion\_Reservation

createtableClient(

idClientsmallintprimarykey,

nomCLientvarchar(30),

adresseClientvarchar(200),

telClientvarchar(15))

go

createtableReservation(

codeReservationsmallintprimarykey,

idClientsmallintforeignkeyreferencesclient,

dateRessmalldatetime,

pensionCompletebit,

tauxReductionint)

go

createtableSejour (

numSejoursmallintprimarykey,

codeReservationsmallintforeignkeyreferencesreservation,

dateSejoursmalldatetime,

typeSejourvarchar(30),

dureeSejoursmallint)

go

createtablereservationAnnulee(

codeReservationsmallintforeignkeyreferencesreservation,

idClientsmallintforeignkeyreferencesclient,

dateAnnulationsmalldatetime,

primarykey (codeReservation,idClient))

1. **Remplir les tables par un jeu d’essai (au moins 2 lignes par table). (4 pts)**

insertintoclientvalues

(10,'sadouk taha','j5 CYM','0600111111'),

(11,'falah mouna','FATH CYM','0600222222');

deletereservation

insertintoreservationvalues

(100,10,'1/11/2017',1,0),

(101,11,'21/12/2017',1,10),

(102,10,'27/12/2017',0,15),

(103,11,'29/12/2017',0,0);

insertintosejourvalues

(1000,100,'10/1/2018','journalier',2),

(1001,101,'10/2/2018','voyage de nices',7),

(1002,102,'10/3/2018','voyage de nices',15),

(1003,103,'15/3/2018','long',15);

insertintoreservationAnnuleevalues

(100,10,'1/12/2017'),(101,11,'1/1/2018');

1. **Ajouter une contrainte à la table Réservation concernant la colonne TauxReduction :**

**Le taux doit être compris entre 0 et 75. (4 pts)**

altertablereservationaddconstraintch\_taux

check(tauxreductionbetween **0** and **75**);

1. **Ecrire une requête qui augmente de 20 % le taux de réservation du mois en cours (attention l’année doit être l’année actuelle). (5 pts)**

updatereservationsettauxReduction+=20

whereMONTH(dateRes)=MONTH(GETDATE())and

YEAR(dateRes)=YEAR(GETDATE());

1. **Ecrire une requête qui liste toutes les réservations par type de séjour et qui ne concerne que les mois prochains. (5 pts)**

select\*fromsejourwhere

((MONTH(dateSejour)>MONTH(GETDATE())AND YEAR(dateSejour)=YEAR(GETDATE())) OR YEAR(dateSejour)>YEAR(GETDATE()))

orderbytypeSejour;

1. **Créer une procédure stockée qui permet de lister les réservations faites par un client donné et retourner leur nombre.**

**Proposer un jeu d’essai correspondant. (5 pts)**

createprocedurelist\_reserv

@idsmallint,@countsmallintoutput

as

set@count= (selectcount(\*)fromreservationwhereidclient=@id)

go

-- appel

declare@nbsmallint

execlist\_reserv 10,@nboutput

select@nb[nombre des reservation]

1. **Créer une fonction qui retourne le nombre des réservations annulées par un client donnée.**

**Proposer un jeu d’essai correspondant. (5 pts)**

createfunctionreserv\_annule (@idsmallint)

returnsint

as

begin

declare@countsmallint

select@count=count(\*)fromreservationAnnuleewhereidclient=@id

return@count

end

--

declare@ressmallint

exec@res=reserv\_annule 10

select@res

1. **Créer un déclencheur qui permet de vérifier lors d’insertion ou une mise à jour dans la table ReservationAnnulee** **que la date d’annulation est supérieure à la date de la réservation correspondante. (5 pts)**

createtriggerverifieronreservationAnnulee

afterinsert,update

as

ifexists(select\*frominsertedIwhere

dateAnnulation<= (selectdatResfromresevationwhere

codeReservation=I.codeReservation))

begin

raiserror('la date de annulation est inferieur a la date de reservation',16,10)

rollback

end

go

1. **Sauvegarder votre base de données dans le chemin suivant : « C:\efmR\_8 » (4 pts)**

backupdatabasegestion\_Reservation

TODISK='C:\efmR\_8\gestion\_Reservation';