# Correction TP2

## 1 PREMIERE ETAPE : CREATION ET MODIFICATION D'ATTRIBUTS

### 1.1. Ajout de colonne

On souhaite intégrer dans la relation *planning* un deuxième tarif de voyage pour les enfants. Effectuer l'ajout de cet attribut *tarifenf* dans la relation en utilisant le même type de données que pour la colonne *tarif*. De manière similaire, le nombre d'enfants doit être ajouté dans la relation *reservation*. Donnez à cet attribut *nbenf* le même type que celui de *nbpers*.

```
ALTER TABLE planning ADD tarifenf numeric(6,2) ;
ALTER TABLE reservation ADD nbenf numeric(2,0) ;
```

### 1.2. Modification de type d'une colonne

On souhaite étendre la définition du type de l'attribut *libelle* de la table *optionv*, en augmentant la taille du nombre de caractères admissibles de 10 caractères. Consultez la définition de cet attribut puis effectuez la modification correspondante et vérifiez-la.

```
ALTER TABLE optionv ALTER libelle  TYPE VARCHAR(30) ;
```

### 1.3. Contrainte de vérification de domaine

1. On souhaite intégrer dans la base une contrainte de domaine pour l'attribut *categorie* de la table *client* qui prendra ses valeurs dans l'ensemble suivant : {PRIVILEGIE, BON, MAUVAIS}. Effectuez cet ajout de contrainte.

```
CREATE DOMAIN DOM_cat_cli AS varchar(15)
CONSTRAINT DOM_CATEGORIE
      CHECK (VALUE IN ('PRIVILEGIE', 'BON', 'MAUVAIS')) ;
```

*Ensuite il faut modifier le type de la colonne CATEGORIE de la table CLIENT*

```
ALTER TABLE CLIENT
ALTER CATEGORIE TYPE DOM_cat_cli;
```

2. Modifier la valeur par défaut de la colonne l'attribut *categorie* de la table *client* pour qu'elle soit MAUVAIS

```
ALTER table CLIENT
ALTER categorie SET DEFAULT 'MAUVAIS';

Update CLIENT
Set categorie ='MAUVAIS'
Where categorie is NULL;
```

3. Spécifiez une contrainte de domaine pour l'attribut *nbetoiles* dans la table *voyage* afin que ses valeurs appartiennent à l'intervalle [1 .. 5].

```
ALTER TABLE VOYAGE
ADD CONSTRAINT DOM_NBETOILES CHECK (NBETOILES BETWEEN 1 AND 5) ;
```

### 1.4. Ajout des tables *capacite* et *hotel*

```
CREATE TYPE type_chambre AS ENUM ('SINGLE','DOUBLE','DOUBLE LUXE',
 'SUITE','SUITE JUNIOR','SUITE PRESTIGE');

CREATE TABLE capacite
(  id_hotel integer,
   typechambre type_chambre,
   nbrch NUMERIC(3,0),
   CONSTRAINT PK_CAPACITE PRIMARY KEY(id_hotel,typechambre)) ;

CREATE TABLE HOTEL
(  id_hotel SERIAL primary key,
   nomHotel VARCHAR(20) default NULL,
   NbEtoiles NUMERIC(1,0) default NULL,
   VilleHotel  VARCHAR(20) default NULL,
   CONSTRAINT DOM_NBETOILES CHECK (NBETOILES BETWEEN 1 AND 5)) ;
```

```
-- création de la clé étrangère dans la table capacite
     ALTER TABLE capacite
     ADD    CONSTRAINT    FK_hotel    FOREIGN    KEY(id_hotel)    REFERENCES
          HOTEL(id_hotel) ;

--  ajout de la clé étrangère dans la table voyage
     ALTER TABLE voyage
     ADD id_hotel integer,
     ADD   CONSTRAINT   FK_Voy_hotel   FOREIGN   KEY(id_hotel)   REFERENCES
          HOTEL(id_hotel) ;

-- transfère des données da la table VOYAGE vers dans la table HOTEL
-- On ne s'occupe pas de la colonne id_hotel qui est auto incrémenté (SERIAL)
     INSERT INTO hotel(nomHotel, NbEtoiles,VilleHotel)
     SELECT  DISTINCT hotel, nbetoiles,villearr
     FROM voyage

-- Valorisation de la colonne id_hotel de la table voyage, on utilise ici une
-- sous requête corrélée dans le update
     UPDATE VOYAGE V
     SET id_hotel=(SELECT id_hotel
                   FROM hotel
                   WHERE nomhotel=V.hotel AND villehotel=V.villearr)

-- vérification des valeurs avant de supprimer les colonnes de la table voyage
     SELECT hotel=nomhotel as "hotel", villearr=villehotel as "ville"
     FROM voyage V JOIN hotel H ON H.id_hotel=V.id_hotel

     La comparaison dans le select renvoie TRUE si le nom des hôtels est correcte,
     FALSE le cas contraire, idem pour la ville, cela permet une vérification
     visuelle rapide, on peut faire plus rapide :

     SELECT *
     FROM voyage V JOIN hotel H ON H.id_hotel=V.id_hotel
     WHERE NOT (hotel=nomhotel  AND villearr=villehotel)

     Ici, si la requête ne retourne aucune ligne, c'est que tout est bon !

-- on peut maintenant supprimer les colonnes de la table voyage
     ALTER TABLE voyage
           DROP villearr,
           DROP hotel,
           DROP nbetoiles ;

-- insertion des données dans la table capacite
-- On peut utiliser une table temporaire ou directement faire des insert
-- avec sous requête, je vais utiliser une table temporaire

     CREATE TABLE temp_capacite
     (  hotel VARCHAR(20),
        typechambre type_chambre,
        nbrch NUMERIC(3,0)) ;

     -- pour être compatible avec la colonne typechambre de la table capacite
```

```
-- insertion des données des hotels
INSERT INTO temp_capacite values
       ('ANTIQUE','SINGLE', 10),('ANTIQUE','DOUBLE',75),
       ('ANTIQUE','SUITE', 5),('ATLAS','DOUBLE',83),
       ('ATLAS','SUITE',27),('OLD BRIDGE','SINGLE',25),
       ('OLD BRIDGE','DOUBLE',75),('SAFARI JAMBO','SINGLE',32),
       ('SAFARI JAMBO','DOUBLE',100),('TRANSATLANTIQUE','DOUBLE',200),
       ('BAMBURI','DOUBLE',150),('BELLERIVE','DOUBLE',56),
       ('BELLERIVE','SUITE',12),('DAR AL BAHAR','SINGLE',10),
       ('DAR AL BAHAR','DOUBLE',12),('EL ALMARANTE','SINGLE',34),
       ('EL ALMARANTE','SUITE',45),('EL ANDALOUS','SINGLE',67),
       ('EL ANDALOUS','DOUBLE',54),('ELIAS BEACH','SINGLE',87),
       ('ELIAS BEACH','DOUBLE',33),('ESPADON','SINGLE',34),
       ('ESPADON','DOUBLE',45),('HASNA','SINGLE',32),
       ('HASNA','DOUBLE',24),('MONDIAL','SINGLE',12),
       ('MONDIAL','DOUBLE',13),('KENZI CLUB','SINGLE',45),
       ('KENZI CLUB','DOUBLE',67) ;
```

**On peut aussi utiliser l'import du fichier csv..**

```
-- il suffit ensuite de travailler sur cette table pour l'insertion dans la
-- table capacite, on utilise encore une sous requête corrélée

INSERT INTO capacite
SELECT (SELECT id_hotel FROM hotel WHERE nomhotel=T.hotel), typechambre  , nbrch
FROM temp_capacite T
```

ATTENTION si vous avez créé un type énuméré pour le type de chambre vous devez *caster* la colonne *typechambre* de la table *temporaire*
Ici je ne le fais pas car quand je créé la table temporaire, j'applique le type *type_chambre* à la colonne *typechambre*

```
-- on supprime la table temporaire
DROP TABLE temp_capacite
```

## 2 DEUXIEME ETAPE : MISES A JOUR DES DONNEES

1. Mettez à jour *l'attribut nbenf* en lui affectant la valeur 2 pour toutes les réservations du client 2103. Modifiez cet attribut en lui donnant la valeur 1 pour les réservations du client *Thomas Jarolim*.

```
UPDATE reservation
SET nbenf = 2
WHERE numcl = 2103 ;

UPDATE reservation
SET nbenf = 1
WHERE numcl IN (SELECT numcl FROM client WHERE nom = 'JAROLIM'
                AND prenom = 'THOMAS') ;
```

2. Pour tous les tuples de la relation *planning*, effectuez la modification de *tarifenf* en lui affectant la valeur de *tarif* réduite de moitié.

```
UPDATE planning SET tarifenf = TARIF / 2 ;
```

3. Insérez deux nouveaux tules dans la relation *client* en vérifiant la mise en œuvre de la contrainte de domaine sur *categorie*. Créez une réservation pour un de ces clients et un voyage existant.

```
INSERT INTO client (numcl, nom, prenom, ville)
VALUES ((select max(numcl)+ 1 from client),
            'ALLEMAND', 'GREGORY', 'AIX-EN-PROVENCE') ;
INSERT INTO client
VALUES ((select max(numcl)+ 1 from client),
            'BENSALAH', 'MALEK',NULL, NULL, 'MARSEILLE', NULL) ;
```

```
--L'insertion suivante échoue à cause de la CI de domaine
INSERT INTO client
VALUES((select max(numcl)+ 1 from client), 'BEUF', 'FRANCOIS',NULL,
NULL, 'MARSEILLE', 'PREMIER') ;
```

**Erreur SQL :**

```
ERREUR:  la nouvelle ligne viole la contrainte de vérification « client » de la
relation « dom_categorie »
DETAIL:  La ligne en échec contient (2307, BEUF, FRANCOIS, null, null, MARSEILLE,
PREMIER)
```

**En effet PREMIER n'est pas une valeur possible définit dans la contrainte de domaine DOM_cat_cli défini au 1.3.**

4. Mettez à jour la relation voyage en tentant de donner une valeur invalide à l'attribut nbetoiles.

```
UPDATE VOYAGE SET NBETOILES = 9 ;
```

**Erreur SQL :**

```
ERREUR:  la nouvelle ligne viole la contrainte de vérification « voyage » de la
relation « dom_nbetoiles »
DETAIL:  La ligne en échec contient (867.00, LISBONNE, PORTUGAL, MARSEILLE,
MONDIAL, 9, 2)
```