

1 CREATION DU SCHEMA DE LA BASE DE DONNEES GITES DE FRANCE

```
-- création du schéma de données
CREATE SCHEMA tp_gite_de_france;
SET search_path=tp_gite_de_france;

-- structure de la table chambres
CREATE TABLE chambres (
    numch          serial primary key,
    numgite        int not null,
    nblits         int not null default 1,
    nblitdo        int default null,
    superficiech   int default null
);

-- structure de la table louer
CREATE TABLE louer (
    numgite        int not null,
    numcli         int not null,
    duree          int default 1,
    nbper          int default 1,
    date_debut     timestamp default current_timestamp,
    constraint pk_louer primary key (numgite,numcli)
);

-- structure de la table clients
CREATE TABLE clients (
    numcli         serial primary key ,
    nomcli         varchar(35) not null,
    adrccli        varchar(50) not null,
    cpcli          decimal(5,0) not null,
    villecli       varchar(35) not null,
    tel            varchar(20) unique not null,
    nb_enfants     int not null default 0
);

-- structure de la table facturer
CREATE TABLE facturer (
    saison         varchar(10) not null,
    numgite        int not null,
    constraint pk_facturer primary key (saison,numgite)
);

-- structure de la table gites
CREATE TABLE gites (
    numgite serial primary key,
    nomgite        varchar(25) default null,
    adrgite        varchar(50) default null,
    cpgite         decimal(5,0) default null,
    villegite      varchar(35) default null,
    telgite        int unique default null,
    descr          varchar(20) not null
);

-- structure de la table saison
CREATE TABLE saison (
    saison varchar(10) primary key
);

-- contraintes pour la table chambres
ALTER TABLE chambres
    ADD CONSTRAINT chambres_ibfk_1 FOREIGN KEY (numgite)
        REFERENCES gites(numgite) ON UPDATE CASCADE;
```

```
-- contraintes pour la table facturer
ALTER TABLE facturer
  add CONSTRAINT facturer_ibfk_1 FOREIGN KEY (saison)
      REFERENCES saison (saison) ON UPDATE CASCADE,
  ADD CONSTRAINT facturer_ibfk_2 FOREIGN KEY (numgite)
      REFERENCES gites (numgite) ON UPDATE CASCADE;

-- contraintes pour la table louer
ALTER TABLE louer
  ADD CONSTRAINT louer_ibfk_1 FOREIGN KEY (numgite)
      REFERENCES gites (numgite) ON UPDATE CASCADE,
  ADD CONSTRAINT louer_ibfk_2 FOREIGN KEY (numcli)
      REFERENCES clients (numcli) ON UPDATE CASCADE;
```

2 MODIFICATION DU DIAGRAMME DE CLASSES

```
-- table: proprietaire
CREATE TABLE proprietaire (
  numprop      serial      primary key,
  nomprop      varchar(35) null,
  adrprop      varchar(50) null,
  cpprop       decimal(5)  null,
  villeprop    varchar(50) null,
  telprop      varchar(20) null
);

-- table : activites
CREATE TABLE activites (
  numact      serial      not null,
  nomact      varchar(45)  null,
  CONSTRAINT pk_activites primary key (numact)
);

-- table : organisme
CREATE TABLE organisme (
  id_orga      serial      not null,
  nom_orga     varchar(45)  not null,
  adr_orga     varchar(175) not null,
  ville_orga   varchar(75)  not null,
  cp_orga      decimal(5,0) not null,
  CONSTRAINT pk_organisme primary key (id_orga)
);

--
-- creation d'un domaine de valeur pour la colonne interet du type composé
--
CREATE DOMAIN dom_interet AS int
  DEFAULT 1
  CONSTRAINT ck_interet check(value between 1 and 5);

--
-- creation d'un type composé pour la colonne info de proposer
--
CREATE type information AS
(
  distance real,
  interet  dom_interet
);

-- table : proposer
CREATE TABLE proposer (
  numgite      int      not null,
  numact       int      not null,
  info         information null,
  CONSTRAINT pk_proposer primary key (numgite, numact));
```

```
-- table : proposer_par
CREATE TABLE proposer_par (
    numgite          int          not null,
    numact           int          not null,
    id_orga          int          not null,
    CONSTRAINT pk_proposer_par PRIMARY KEY (numgite, numact ,id_orga)
);

-- clé étrangère de la table proposer
ALTER TABLE proposer
    ADD CONSTRAINT fk_proposer_activite FOREIGN KEY (numact)
        REFERENCES activites (numact) ON UPDATE CASCADE,
    ADD CONSTRAINT fk_proposer_gite FOREIGN KEY (numgite)
        REFERENCES gites (numgite) ON UPDATE CASCADE;

-- clé étrangère de la table proposer_par
-- ATTENTION à la clé étrangère composée de deux colonnes qui fait
-- référence à la table proposer
ALTER TABLE proposer_par
    ADD CONSTRAINT fk_pro_org FOREIGN KEY (id_orga)
        REFERENCES organisme (id_orga) ON UPDATE CASCADE,
    ADD CONSTRAINT fk_pr FOREIGN KEY (numgite, numact)
        REFERENCES proposer (numgite, numact) ON UPDATE CASCADE;

-- ajout de l'attribut clé étrangère et de la clé étrangère
-- dans la table gites
ALTER TABLE gites
    add numprop int not null,
    add FOREIGN KEY (numprop)
        REFERENCES proprietaire(numprop) ON UPDATE CASCADE;

-- ajout des attributs dans la table gites
ALTER TABLE gites
    add superficiegit int null,
    add nbepis int default 1,
    add nbpers int default 2;

-- ajout de l'attribut dans la table facturer
ALTER TABLE facturer
    add tarifttc decimal(6,2) not null;
```

3 SUPPRESSION D'ATTRIBUTS

```
ALTER table CLIENTS drop COLUMN NB_ENFANTS;
```

4 MODIFICATION DU TYPE D'UN ATTRIBUT

```
ALTER table GITES
alter column TELGITE TYPE varchar(15),
add constraint ck_tel check(TELGITE SIMILAR TO '[0-9][0-9]-[0-9][0-9]-[0-9][0-9]-[0-9][0-9]-[0-9][0-9]-[0-9][0-9]');
```

5 MODIFICATION DU NOM ET DU TYPE D'UN ATTRIBUT

```
-- renommer la table
ALTER TABLE GITES
    RENAME COLUMN DESCR TO description;

-- modifier le type
ALTER TABLE GITES
    ALTER COLUMN description TYPE VARCHAR(255);
```

6 AJOUT D'ENREGISTREMENTS

A. Ajout d'un nouveau gîte :

ATTENTION : à l'ordre dans lequel vous allez enregistrer les informations dans la base de données, il est important de commencer par les tables qui n'ont pas de clé étrangère :

1. PROPRIETAIRE : coordonnées du propriétaire
2. GITES : adresse du gîte, superficie, nombre de personnes et le nombre d'épis
3. CHAMBRES : caractéristiques de chaque chambre + le lien vers MAISON_INDIV
4. SAISON : ajout des saisons (elles seront ajoutées une seule fois)
5. FACTURER : le tarif pour chaque saison

```
-- AJOUT DU PROPRIETAIRE
INSERT INTO proprietaire
VALUES (default, 'dusic pierre', '44, rue pierre curie',
       94140, 'Alfortville', '01-45-02-10-20');

-- AJOUT DU GITE DANS LA TABLE GITES
INSERT INTO gites
VALUES (default, 'les castors', '5 rue qui monte', 74100, 'st gervais',
       '04-42-05-70-52',
       'charmant gite de plein pied au cœur de la montagne',
       (select max(numprop) from proprietaire), 120, 3, 10);

-- AJOUT DES CHAMBRES ET DE LEURS CARACTERISTIQUES DANS LA TABLE CHAMBRES
-- la sous requête permet de récupérer l'identifiant du gite précédemment créé,
-- pour associer la chambre à un gite

-- Chambre 1 : 1 lit simple, 1 lit double, superficie de 15m²;
INSERT INTO Chambres VALUES (default, (select max(numgite) from gites), 1, 1, 15);

-- Chambre 2 : 1 lit double, superficie de 20m²;
INSERT INTO Chambres VALUES (default, (select max(numgite) from gites), 0, 1, 20);

-- AJOUT DES SAISONS DANS LA TABLE SAISON
INSERT INTO saison VALUES ('BASSE');
INSERT INTO saison VALUES ('MOYENNE');
INSERT INTO saison VALUES ('HAUTE');

-- AJOUT DES TARIFS DANS LA TABLE FACTURER
-- la sous requête permet de récupérer l'identifiant du gite précédemment créé,
-- pour associer les prix par saison à un gite
INSERT INTO facturer VALUES ('BASSE', (select max(numgite) from gites), 800);
INSERT INTO facturer VALUES ('MOYENNE', (select max(numgite) from gites), 1200);
INSERT INTO facturer VALUES ('HAUTE', (select max(numgite) from gites), 1800);
```

B. Ajout d'une réservation :

Le client Carmignac Laurent habitant au 4, allée des Camélias 13080 Luynes, téléphone 04-42-50-65-52, loue le nouveau gîte du 5/04/2005 pour une durée de 15 jours, cette location concerne 7 personnes

```
-- AJOUT DU CLIENT DANS LA TABLE CLIENTS
INSERT INTO clients VALUES (default, 'Carmignac Laurent',
                             '4, allée des Camélias', 13080,
                             'Luynes', '04-42-50-65-52');

-- AJOUT DES CARACTERISTIQUES DE LA RESERVATION DANS LA TABLE LOUER
INSERT INTO louer VALUES ((select max(numgite) from gites),
                           (select max(numcli) from clients), 15, 7, '2005-04-05');
```

Ecrire les requêtes SQL qui permettent de vérifier que les insertions se sont bien effectuées.

```
-- vérifie l'insertion du gite
SELECT *
FROM gites G JOIN chambres C ON C.numgite=G.numgite

-- vérifie l'insertion des tarifs
SELECT *
FROM gites G JOIN facturer F ON G.numgite=F.numgite

-- vérifie l'insertion de la location
SELECT *
FROM gites G JOIN louer L ON G.numgite=L.numgite
JOIN clients C ON L.numcli=C.numcli
```

C. Ajout d'une nouvelle réservation :

Le même client que précédemment loue le même gîte du 2/07/2005 pour une durée de 3 semaines, cette location concerne 10 personnes

```
-- AJOUT DES CARACTERISTIQUES DE LA NOUVELLE RESERVATION DANS LA TABLE LOUER
INSERT INTO louer VALUES((select max(numgite)from gites),
(select max(numcli)from clients),21,10,'2005-07-02');
```

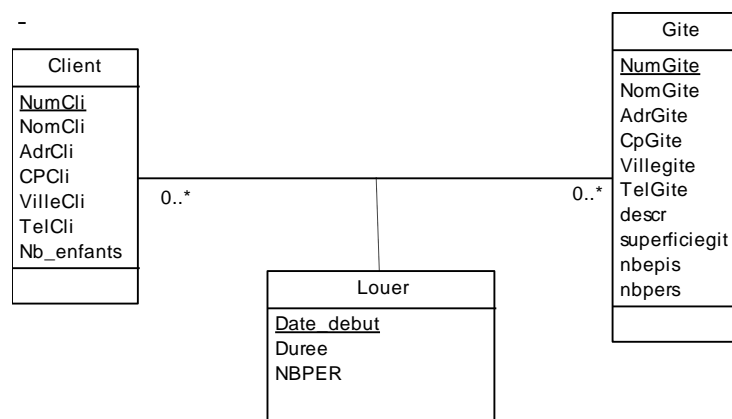
Il y a une erreur ?????

Erreur SQL :

ERREUR: la valeur d'une clé dupliquée rompt la contrainte unique « pk_louer »
DETAIL: La clé « (numgite, numcli)=(1, 1) » existe déjà.

Effectivement il est impossible pour un même client de louer le même gîte et ce quel que soit les dates, car la clé primaire de la table LOUER n'est composée que de l'identifiant de la table GITES et de celui de la table CLIENTS.

Il faut donc que la date de début de location (louer) soit clé primaire de la table LOUER, pour cela on modifie le diagramme de classes :



Dans la base de données, il suffit de supprimer la contrainte de clé primaire dans la table LOUER et d'en ajouter une nouvelle qui sera composée des id des tables GITES, CLIENTS et LOUER,

```
-- suppression de la cle primaire de la table louer
alter table louer drop constraint pk_louer;

-- ajout de la nouvelle la cle primaire de la table louer
alter table louer
add primary key (numgite, numcli, date_debut);
```

Vous pouvez maintenant ajouter l'enregistrement de la location

```
INSERT INTO louer VALUES((select max(numgite)from gites),
(select max(numcli)from clients),21,10,'2005-07-02');
```

7 AJOUT D'UNE ACTIVITE

Il faut associer l'activité *alpinisme* au gite précédemment saisi, cette activité se situe à 23 km du gite, l'intérêt est de 4 sur une échelle de 5.

Il y a deux organismes qui proposent cette activité :

Bureau des guides de Chamonix 190 Place de l'Église, 74400 Chamonix-Mont-Blanc

Les Guides Indépendants de Chamonix, 45 rue blanche, 74400 Chamonix-Mont-Blanc

```
-- ajout de l'activité
INSERT INTO activites VALUES(default, 'alpinisme');

-- ajout du lien entre l'activité et le gite(table proposer)
-- ATTENTION à l'insertion dans un type composé
INSERT INTO proposer VALUES((select max(numgite)from gites),
                             (select max(numact)from activites),
                             ROW(23,4));

-- ajout des organismes
INSERT INTO organisme VALUES(default, 'Bureau des guides de Chamonix',
                              '190 Place de l'Église','Chamonix-Mont-Blanc',
                              74400);
INSERT INTO organisme VALUES(default, 'Les Guides Indépendants de Chamonix',
                              '45 rue blanche','Chamonix-Mont-Blanc',74400);

-- ajout des liens entre les organisme et la table proposer
INSERT INTO proposer_par VALUES( (select max(numgite)from gites),
                                 (select max(numact)from activites),
                                 (select max(id_orga)from organisme));

INSERT INTO proposer_par VALUES( (select max(numgite)from gites),
                                 (select max(numact)from activites),
                                 (select min(id_orga)from organisme));

-- Verification des insertions

SELECT nomgite, nomact,nom_orga, (info).distance,(info).interet
FROM activites A JOIN proposer P ON P.numact=A.numact
JOIN gites G ON G.numgite=P.numgite
JOIN proposer_par Pp ON Pp.numgite=P.numgite
AND Pp.numact=P.numact
JOIN organisme O ON o.id_orga=Pp.id_orga
```