

TRAVAIL A FAIRE

Q1. CREATION DE VUES

A. Vue clients et gites

Créer une vues, que vous nommerez *view_location*, qui permet de voir les clients et les gites qu'ils ont loués, afficher le résultat de la vue créée.

```
CREATE OR REPLACE VIEW view_location (numcl, nom_cl, duree_loc, nbper,
date_loc, numgite, nomgite)
```

```
AS
```

```
select C.numcli,nomcli,duree,nbper,date_debut,G.numgite,nomgite
from clients C join louer L on C.numcli=L.numcli
      join gites G on G.numgite=L.numgite
order by C.numcli;
```

-- affichage du résultat de la vue

```
select *
from view_location;
```

numcl	nom_cl	duree_loc	nbper	date_loc	numgite	nomgite
1	Pruitt	7	2	2011-01-19	1	Le Castor
2	Anderson	13	11	2011-09-27	3	Belle pente
2	Anderson	8	4	2013-08-27	3	Belle pente
3	Perry	11	2	2011-09-15	4	La belle bosse
6	Ramirez	7	7	2011-05-29	5	l'Edelweiss
7	Huff	6	12	2014-09-15	4	La belle bosse
7	Huff	12	7	2014-03-08	4	La belle bosse

44 lignes

B. Vue gites, propriétaire et tarif

Créer une vue, que vous nommerez *view_tarif_gite*, qui permet de voir les gites, leur propriétaire et les tarifs en fonction de la saison, afficher le résultat de la vue créée

```
CREATE OR REPLACE VIEW view_tarif_gite (numgite, nom_gite,
nom_proprietaire, saison, tarif)
```

```
AS
```

```
select G.Numgite,nomgite,nomprop, saison, tarifttc
from gites G join facturer F on G.numgite=F.numgite
      join proprietaire P ON P.numprop=G.numprop
order by g.numgite;
```

-- affichage du résultat de la vue

```
select *
from view_tarif_gite;
```

numgite	nom_gite	nom_proprietaire	saison	tarif
1	Le Castor	Finn	HAUTE	600.00
1	Le Castor	Finn	BASSE	387.00
1	Le Castor	Finn	MOYENNE	450.00
2	Le col blanc	Clinton	BASSE	300.00
2	Le col blanc	Clinton	HAUTE	700.00
2	Le col blanc	Clinton	MOYENNE	400.00

51 lignes

C. Vue gites jamais loués

Créer une vue, que vous nommerez *view_gite_sans_loc*, qui permet de voir les gites jamais loués, afficher le résultat de la vue créée

```
CREATE OR REPLACE VIEW view_gite_sans_loc (num_gite, nom_gite,
nom_proprietaire, saison, tarif, tel_gite, description,num_prop,
superficie_gite,nb_epis,nb_personne)
AS
SELECT numgite, nomgite,nomprop, saison, tarif,ttc,telgite, description,
numprop, superficiegit, nbepis, nbpers
FROM gites JOIN proprietaire using(numprop)
JOIN facturer using(numgite)
WHERE numgite NOT IN(SELECT numgite FROM louer);

-- affichage du résultat de la vue
select *
from view_gite_sans_loc;
```

num_gite	nom_gite	nom_proprietaire	saison	tarif	tel_gite	description	num_prop	superficie_gite	nb_epis	nb_personne
2	Le col blanc	Clinton	BASSE	300.00	07-84-59-55-17	Lorem ipsum dolor	14	NULL	2	10
2	Le col blanc	Clinton	HAUTE	700.00	07-84-59-55-17	Lorem ipsum dolor	14	NULL	2	10
2	Le col blanc	Clinton	MOYENNE	400.00	07-84-59-55-17	Lorem ipsum dolor	14	NULL	2	10
6	Nevada	Quamar	HAUTE	800.00	07-47-05-64-09	Lorem ipsum dolor sit amet, consectetur adipisci...	11	NULL	1	9
6	Nevada	Quamar	BASSE	450.00	07-47-05-64-09	Lorem ipsum dolor sit amet, consectetur adipisci...	11	NULL	1	9
6	Nevada	Quamar	MOYENNE	600.00	07-47-05-64-09	Lorem ipsum dolor sit amet, consectetur adipisci...	11	NULL	1	9
11	Basil	Carly	HAUTE	1355.00	05-74-51-48-45	Lorem ipsum dolor	7	NULL	1	5
11	Basil	Carly	BASSE	850.00	05-74-51-48-45	Lorem ipsum dolor	7	NULL	1	5
11	Basil	Carly	MOYENNE	1050.00	05-74-51-48-45	Lorem ipsum dolor	7	NULL	1	5
17	la camelle	Chancellor	BASSE	350.00	06-83-42-07-43	Lorem ipsum dolor sit amet, consectetur	17	NULL	1	13
17	la camelle	Chancellor	HAUTE	750.00	06-83-42-07-43	Lorem ipsum dolor sit amet, consectetur	17	NULL	1	13
17	la camelle	Chancellor	MOYENNE	550.00	06-83-42-07-43	Lorem ipsum dolor sit amet, consectetur	17	NULL	1	13

D. Vue gites et activités

Créer une vues qui permet de voir les gites et les activités associées
Attention à la colonne info de la table *proposer*

```
CREATE OR REPLACE VIEW view_gite_activite (numgite, nom_gite,
activite,distance,interet)
AS
select G.numgite,nomgite,nomact,(info).distance, (info).interet
from gites G join proposer P on P.numgite=G.numgite
join activites A on A.numact=P.numact;

-- affichage du résultat de la vue
select *
from view_gite_activite;
```

numgite	nom_gite	activite	saison	tarif	interet
13	Germaine	3	Ski	13	3
13	Germaine	1	alpinisme	13	1
13	Germaine	2	Randonnée	13	2
13	Germaine	5	Escalade	13	5
15	Fletcher	1	alpinisme	15	1
15	Fletcher	6	VTT	15	2
15	Fletcher	3	Ski	15	3

29 lignes

Q2. CREATION DE REGLES

A. règle sur un update

Créer une règle sur la vue view_location qui lors de la mise à jours de la durée de location (duree_loc) et/ou le nombre de personnes (nbper) met à jours la table correspondante (louer), *La mise à jour ne se fera que si le nombre de personnes et inférieur ou égal au nombre de personnes défini dans le gite*
testez cette règle

```
CREATE OR REPLACE RULE update_view_location
AS ON UPDATE TO view_location
DO INSTEAD
UPDATE louer
SET duree = NEW.duree_loc,
    nbper = NEW.nbper
WHERE numgite = NEW.numgite
    AND numcli = NEW.numcli
    AND date_debut=NEW.date_loc
    AND NEW.nbper <=(SELECT nbpers FROM gites WHERE numgite= NEW.numgite);
```

-- test

```
update view_location
set duree_loc=3,
    nbper=3
where numgite=1 and numcli=1 and date_loc='2011-01-19';
-- mise à jour possible nbper <= 4 : 1 ligne mises à jours

update view_location
set nbper=2
where numgite=1;
-- Les autres critères ne sont pas pris en compte : 4 lignes mises à jours

update view_location
set nbper=9
where numgite=1 ;
-- pas de mise à jours nbper > le nb pers du gite 1
```

B. règle sur un delete

Lors de la suppression d'un gite via la vue view_location, il faut supprimer les lignes des différentes tables qui font référence au gite à supprimer !!

```
CREATE or replace RULE delete_view_location
AS ON delete TO view_location
DO INSTEAD
(
DELETE from chambres
WHERE numgite = OLD.numgite;

DELETE from facturer
WHERE numgite = OLD.numgite;

DELETE from proposer_par
WHERE numgite = OLD.numgite;

DELETE from proposer
WHERE numgite = OLD.numgite;

DELETE from louer
WHERE numgite = OLD.numgite;

DELETE from gites
WHERE numgite = OLD.numgite; );
```

-- test de la règle

```
DELETE FROM view_location
WHERE numgite=1;

SELECT *
FROM gites NATURAL JOIN louer
      NATURAL JOIN chambres
      NATURAL JOIN proposer
WHERE numgite=1 ;
```

-- Pas de ligne retournée par Le select

C. Règle sur un insert

Ajouter une règle pour éviter une erreur lors de l'insertion si doublon sur une pk.

lors de l'insertion d'un nouveau gite (table gite) si la valeur de la clé primaire existe il faudra, à la place de l'insertion, mettre à jour le gite avec les données de l'insertion..

```
CREATE OR REPLACE RULE insert_gite AS ON INSERT TO gites
WHERE exists (SELECT * FROM gites WHERE numgite=NEW.numgite)
DO INSTEAD
  UPDATE gites
  SET nomgite=NEW.nomgite,
      adrgite=NEW.adrgite,
      cpgite=NEW.cpgite,
      villegite=NEW.villegite,
      telgite=NEW.telgite,
      description=NEW.description,
      numprop=NEW.numprop,
      superficiegit=NEW.superficiegit,
      nbepis=NEW.nbepis,
      nbpers=NEW.nbpers
  WHERE numgite=NEW.numgite;
```

-- test de la règle avec ajout effectif

```
INSERT INTO gites VALUES(default, 'les castors', '4 rue des castors', 73450,
                          'Valmorel', '06-06-07-08-09', 'bien',16,120,3,5) ;
```

-- test de la règle avec modification de la ligne du gite N°13

```
INSERT INTO gites VALUES(13, 'Germaine', 'CP 344, 5737 Scelerisque Avenue',
                          73607, 'albertville', '04-66-88-83-44', 'Lorem', 1, 150, 2, 6);
```

Q3. MAINTENANCE CORRECTIVE DE LA BDD

Il s'agit de vérifier que le nombre de personnes déclarés par gîte (nbpers) est bien conforme au nombre de couchages, lits simple et double (nblits, nblitdo)

- Écrire une requête qui permet d'afficher :
 - le N° du gîte
 - Le nombre de personnes déclaré dans le gîte
 - Le nombre de lits simple et le nombre lits double
 - et la capacité de couchage par gîte, en fonction du nombre de lit et double (1 place par lit simple et 2 par lit double)

```
SELECT  G.numgite as "N° du gîte",
        SUM (nblits) AS "nb lits simple",
        SUM (nblitdo) AS "nb lits double",
        SUM (nblits*1+ nblitdo*2) AS "capacité de couchage",
        nbpers AS "Nb personnes déclaré",
FROM gites G JOIN chambres C ON G.numgite=C.numgite
GROUP BY G.numgite
HAVING nbpers!= SUM(nblits*1+ nblitdo*2)
ORDER BY 1;
```

N° du gîte	Nb personnes déclaré	nb lits simple	nb lits double	capacité de couchage
7	28	13	7	27
9	10	4	4	12
11	5	3	2	7
12	2	3	1	5
13	6	3	1	5

- Écrire une requête qui permet de mettre en corrélation le nombre de personnes déclaré avec la capacité de couchage (une requête corrélée et nécessaire pour s'en sortir)
Exemple : Si la capacité de couchage est de 6 il faut que le nombre de personne de la table *gites* soit aussi à 6

```
update gites Gi
set nbpers=(SELECT sum(nblits*1+ nblitdo*2)
            FROM chambres where numgite=Gi.numgite)
WHERE nbpers!= (SELECT sum(nblits*1+ nblitdo*2)
               FROM chambres where numgite=Gi.numgite) ;
```