- 1. Example
  - 1. NE MARCHE PAS
- 2. Remarque:
- 3. Sous requete de INSERT, DELETE et UPDATE
  - 1. Example
- 4. Vues
  - 1. Example
    - 1. ATTENTION
  - 2. EXAMPLE

```
OUTER JOIN et condition ON
```

Dans la condition du ON on doit avoir au moins une "vrai condition de jointure

### 0.1. Example

```
r FULL JOIN OUTER JOIN s ON R.a = s.b

Autre condition r.a+r.b = s.c

(r.x || 'a') = s.y (|| concat )
```

#### **NE MARCHE PAS**

- s.a IS NOT DISTINCT FROM r.b
- s.a != r.b

# 1. Remarque:

Cette contrainte n'existe pas pour le INNER JOIN

```
r FULL OUTER JOIN s ON condition
```

le resultat est constitué de - r INNER JOIN s ON condition - puis il ajoute des ligne de r qui n'apparaisse pas (complétées par des NULL ) - idem pour s

# 2. Sous requete de INSERT, DELETE et UPDATE

## 2.1. Example

ON cree une facture n° 100 qui commande au magasin tout les produits en stock dans ce magasin  $\,$ 

```
INSERT INTO facture VALUES(100, 1, 'xxx', NULL, DEFAULT);
```

```
INSERT INTO ligne_commande
   SELECT 100,id_produit,1,NULL
   FROM stock
   WHERE id_magasin = 1;
```

Augmenter de 20% le prix des produits ayant une moyenne de satisfaction de  $\geq$  4

```
UPDATE produits p
   SET prix = 1.2*prix
WHERE (
   SELECT AVG(satisfaction)
   FROM ligne_commande
   WHERE p.id_produit = id_produit
   ) >=4;
```

Supprimer du stock les produits qu'au moins un client a noté 0

```
DELETE FROM stock AS s
WHERE EXISTS (
    SELECT *
    FROM ligne_commande
    WHERE satisfaction = 0
    AND s.id_produit = id_produit
    );
```

# 3. Vues

Une vue est une requête pré-enregisté qui va plus ou moins être utilisé comme une table

## 3.1. Example

```
CREATE VIEW facture_moyenne
AS SELECT no_facture, id_magasin
FROM facture;
```

```
SELECT id_magasin, COUNT(*)
FROM facture_moyenne
GROUP BY id_magasin;
```

Pour calculer cette requête postgres remplace la requête correspondante

Il n'y a aucune donnée dans une table, juste la requête qui l'a créée

#### Interêt:

- Donner un acces limité a certain utilisateur
- Données des vues adaptées a un logiciel
- Dans le cas des requête complexes qui servent souvent a d'autre requêtes. limiter cet usage à des cas particuliers

#### **ATTENTION**

INSERT, UPDATE, ou DELETE peuvent fonctionner sur des vue simple Ces vue ne doivent avoir qu'une seul table dans le FROM

## 3.2. EXAMPLE