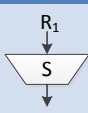
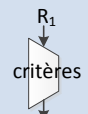
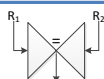


Algèbre relationnel et SQL

BD1 – Chapitre 2

I. Algèbre relationnel

R_{ix} relation de schéma SR_i

Repres.	Fonction	Ecriture	Description	Equivalent SQL
	Projection	$\pi(R_1, S) \quad S \subset SR_1$	Garde les champs S	SELECT champs
	Produit cartésien	$X(R_1, R_2)$	Toutes les combinaisons	FROM R1, R2
	Sélection	$\sigma(R_1, \text{critères})$	Garde les lignes qui satisfont les critères	WHERE critères
	Jointure Semi-jointure à gauche	$\bowtie (R_1, R_2, \text{critères})$ $\ltimes (R_1, R_2, \text{critères})$	X avec critères \ltimes garde le schéma de R_1	FROM R1, R2 WHERE R1.id = R2.id
	Division	$\div (R_1, R_2) \quad SR_2 \subset SR_1$	Sous-tuples de R_1 qui satisfont R_2	
	Union	$\cup (R_{1a}, R_{1b})$		(REQ1) UNION (REQ2)
	Intersection	$\cap (R_{1a}, R_{1b})$		(REQ1) INTERSECT (REQ2)
	Différence	$-(R_{1a}, R_{1b})$		(REQ1) EXCEPT (REQ2)

II. SQL

1. Manipulation de table

```
CREATE TABLE nomTable (
    champNat      INTEGER      UNIQUE BETWEEN a AND b PRIMARY,
    champDec      DECIMAL(m,n) NOT NULL,
    ChampReel    FLOAT,
    champChaine   CHAR(longueur),
    champDate     DATE
)

ALTER TABLE nomTable ADD/REMOVE COLUMN champ TYPE -- ajoute une colonne
DROP TABLE nomTable -- supprime la table
TRUNCATE TABLE nomTable -- vide la table
INSERT INTO nomTable [(champs)] VALUES (val, ...) -- insertion dans la table
DELETE FROM nomTable WHERE critères -- suppression dans la table
UPDATE nomTable SET champ1 = val WHERE ... -- MAJ de la table

CREATE VIEW nomVue(champs) AS (REQ) -- crée une vue
DROP VIEW nomVue -- supprime une vue
```

2. Requête de sélection

```
SELECT [DISTINCT] T1.c1, T2.c1, COUNT/MIN/MAX/SUM/AVG(T1.c2), ...
FROM table1 T1, table2 T2, ...      Produit cartésien des tables
WHERE T1.c1 = <> < > T2.c1      AND/OR      Selection, critères de jointures
    T1.c2 LIKE 'bla%bla'        AND/OR      % : chaine quelconque
    T1.c3 BETWEEN a AND b      AND/OR      ANY / ALL : critère vrai pour au moins un / tous
    T1.c4 IN (a, b, c) / (REQ) AND/OR      les résultats de (REQ2)
    T1.c1 > ANY/ALL (REQ2)      AND/OR      EXISTS : Garde la ligne si (REQ3) est non vide
    [NOT] EXISTS (REQ3)        grâce à cette ligne (REQ3 utilise les champs de
                                cette requête dans son WHERE)
GROUP BY T1.c1, T2.c1 ...      Délimite un groupe (limite COUNT, ...)
HAVING COUNT/...(*) > 1      Conditions sur les groupes du GROUP BY
ORDER BY champ ASC/DESC
```