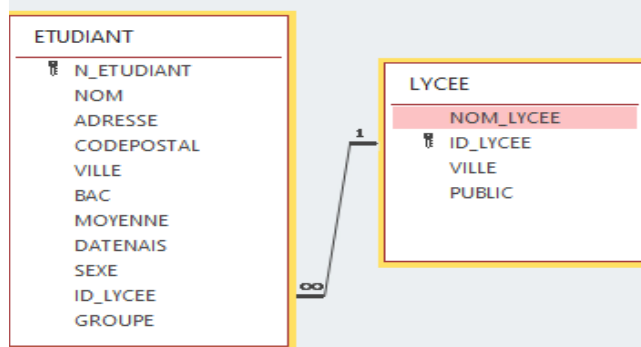


**Présentation.** On utilise ici une base de données (BD) du TP1. Cette BD est Constituée de 2 tables : **Etudiant** et **LYCEE**



**Exercice 1.** Une copie de cette Base de Données le site intranet. Rappelons que les deux table ETUDIANT1 et ETUDIANT2 sont fusionnés dans une seule table ETUDIANT.

Lancer un navigateur web puis récupérer cette BD et l’enregistrer dans le dossier mes documents. Ouvrir la base de données.

**Exercice 2 : Requêtes de recherche avec une partie d’informations**

2.1/ Donner les clients dont le nom ressemble à (le masque de ressemblance sera saisie au clavier).

```
SELECT * FROM ETUDIANT
WHERE (NOM Like "*" & [donner le nom ou une partie du nom] & "*") ;
```

2.3/ Idem que la question 2.1. Donner les clients dont l’adresse ressemble à

2.4/ Idem que la question précédente. Donner les clients dont la ville ressemble à

**Exercice 3 : Requêtes de comptage**

Faire une requête donnant le nombre de clients pour chaque catégorie de bac :

```
SELECT ETUDIANT.BAC, Count (N_ETUDIANT) AS Nombre_etudiants_par_bac FROM
ETUDIANT GROUP BY BAC; Résultat :
```

BAC	Nombre_etudiants_par_bac
ES	2
L	4
S	4
STI	6
STT	4

L’option **AS** permet de choisir un **titre** pour la colonne affichée

**Exercice3.1.** Faire une requête affichant le nombre de clients par ville. Enregistrer sous **tp7r7.5**

**Exercice3.2.** Faire une requête permettant d’afficher le nombre d’tudiants par catégorie d’âge.

Indication : on utilise ici la requête de l’exercice 14 de tp1 (**TP1r14**): `select nom, 2020 - year(datenais) from ETUDIANT1;`

```
SELECT (2020 - year(datenais)) as age, Count (N_ETUDIANT) AS
Nombre_etudiants_par_age FROM ETUDIANT GROUP BY (2020 - year(datenais));
```

Voice le résultat:

age	Nombre_etudiants_par_age
17	2
24	3
25	2
26	2
42	3
43	4
44	2
45	1

#### Exercice 4 : Requêtes multi-tables (Jointure)

Sélectionner des données se trouvant sur plusieurs tables.

Après la clause **FROM** on note les tables source d'informations et après **WHERE** on note les conditions de jointure, qui indiquent en générale des égalités entre clé d'une table et clé étrangère d'une autre table.

Exemple : Faire une requête affichant nom de l'étudiant et le nom de son lycée

```
SELECT NOM, NOM_LYCEE
FROM LYCEE, ETUDIANT
WHERE (LYCEE.ID_LYCEE = ETUDIANT.ID_LYCEE);
```

L'exécution de cette requête donne :

NOM	NOM_LYCEE
AMOUROUX	Saint Joseph
ERNOULT	Saint Joseph
FORD	Saint Joseph
JAMES	Saint Joseph

4.1/ Faire une requête SQL affichant toutes les informations disponibles dans la base sur chaque étudiant.

Indication: `SELECT ETUDIANT.*, LYCEE.* FROM LYCEE, ETUDIANT WHERE (LYCEE.ID_LYCEE = ETUDIANT.ID_LYCEE);`

#### 5. Requêtes de mise à jour, l'instruction UPDATE

Etudiant COLIN :

N_ETUDIANT	NOM	ADRESSE	CODEPOSTAL	VILLE	BAC	MOYENNE	DATENAIS	SEXE	ID_LYCEE	GROUPE
205	COLIN	rue Edouard Larue	76600	LE HAVRE	L	12,975	09/11/1978	F	0760023M	2

N'habite plus dans la rue Edouard Larue sa nouvelle adresse est : Rue FOCH. Voici la requête de mise à jour :

```
UPDATE ETUDIANT SET ADRESSE = "Rue FOCH" WHERE NOM="COLIN" ;
```

Après l'exécution de cette requête l'enregistrement COLIN devient :

N_ETUDIANT	NOM	ADRESSE	CODEPOSTAL	VILLE	BAC	MOYENNE	DATENAIS	SEXE	ID_LYCEE	GROUPE
205	COLIN	Rue FOCH	76600	LE HAVRE	L	12,975	09/11/1978	F	0760023M	2

#### 6. Requêtes de mise à jour, l'instruction DELETE /

Effacer un étudiant MICH0

N_ETUDIANT	NOM	ADRESSE	CODEPOSTAL	VILLE	BAC	MOYENNE	DATENAIS	SEXE	ID_LYCEE	GROUPE
100	MICH0	COTE DU MOULIN	76133	CRICQUETOT-L'ESNEVAL	STI	11,25	08/02/1996	M	0760023M	1

a démissionné pour l'effacer de la bd on utilise la requête suivante :

```
DELETE FROM ETUDIANT WHERE N_ETUDIANT = 100;
```

**6. Trie. La clause ORDER BY** permet de trier les enregistrements sélectionnés par une requête SQL. Les enregistrements sélectionnés peuvent être triés.

**6.1 Exemple1 :** Pour ordonner les étudiants par nom : `SELECT * FROM ETUDIANT order by nom;`

Résultat :

N_ETUDIANT	NOM	ADRESSE	CODEPOSTAL	VILLE	BAC	MOYENNE	DATENAIS	SEXE	ID_LYCEE	GROUPE
101	AMOUROUX	rue Arthur Hulme	76290	ELBEUF	STI	12,617	30/09/1994	M	0761710W	1
203	ARTHUR	rue Clément Ader	76330	NOTRE DAME GRAVENCHON	STT	14,442	07/02/1977	F	0760072R	2
108	ATKINSON	11 RUE GUY MAUPASSANT	76720	GODERVILLE	S	12,958	08/02/1996	M	0760058A	1
102	AUBER	VERDUN	76300	DEVILLE	STI	10,833	28/09/1995	M	0760110G	1
...										

**6.2 Exemple2 :** Pour ordonner les étudiants par bac puis pour le meme bac par nom :

```
SELECT * FROM ETUDIANT order by bac, nom ;
```

Résultat :

N_ETUDIANT	NOM	ADRESSE	CODEPOSTAL	VILLE	BAC	MOYENNE	DATENAIS	SEXE	ID_LYCEE	GROUPE
207	BEN	Le Petit Goulet	76640	FAUVILLE	ES	12,75	07/06/1978	F	0760072R	2
107	SAINT MAXENT	10 RUE DU HAVRE	76640	LILLEBONNE	ES	12,75	28/09/1995	F	0760072R	1

N_ETUDIANT	NOM	ADRESSE	CODEPOSTAL	VILLE	BAC	MOYENNE	DATENAIS	SEXE	ID_LYCEE	GROUPE
105	AVENEL	MARYSE LEBLANC	76600	HARFLEUR	L	12,975	30/09/1994	F	0760023M	1
104	BENSAID	25 RUE DU CARILLON	76430	ANNECY LE VIEUX	L	13,475	16/10/2003	F	0760072R	1
205	COLIN	Rue FOCH	76600	LE HAVRE	L	12,975	09/11/1978	F	0760023M	2
204	JOHN	rue des lupins	76430	LA REMUEE	L	13,475	28/07/1976	F	0760072R	2
108	ATKINSON	11 RUE GUY MAUPASSANT	76720	GODERVILLE	S	12,958	08/02/1996	M	0760058A	1

**6.3 Exercice :** On veut ordonner les élèves par date de naissance . Indication : `SELECT * FROM ETUDIANT ORDER BY (datenaiss) DESC ;` (ordre décroissant)

N_ETUDIANT	NOM	ADRESSE	CODEPOSTAL	VILLE	BAC	MOYENNE	DATENAIS	SEXE	ID_LYCEE	GROUPE
104	BENSAID	25 RUE DU CARILLON	76430	ANNECY LE VIEUX	L	13,475	16/10/2003	F	0760072R	1
106	ERNOULT	38, des Cevennes	76620	Paris	S	12,267	16/10/2003	M	0761710W	1
...	...									
209	DAVID	chemin des vallées	76930	OCTEVILLE SUR MER	STT	12,592	05/03/1976	M	0760023M	a
202	SMITH	D rue du cours	76300	SOTTEVILLE LES ROU.	STI	10,833	20/05/1975	M	0760110G	2

**6.4 Exercice :** On veut ordonner les élèves par lycée et pour le même lycée par ordre alphabétique des noms. Indication : `SELECT ETUDIANT.*, LYCEE.NOM_LYCEE FROM ETUDIANT, LYCEE`

where `LYCEE.ID_LYCEE = ETUDIANT.ID_LYCEE ORDER BY LYCEE.NOM_LYCEE, ETUDIANT.NOM;` **Résultat :**

N_ETUDIANT	NOM	ADRESSE	CODEPOSTAL	VILLE	BAC	MOYENNE	DATENAIS	SEXE	ID_LYCEE	GROUPE	NOM_LYCEE
203	ARTHUR	Clément Ader	76330	NOTRE DAME GRAVENCHON	STT	14,442	07/02/1977	F	0760072R	2	Guill. Le Conquérant
103	AUDEBERT	BEAU JOLAY	76330	LA GUEROUULDE	STT	14,442	08/02/1996	F	0760072R	1	Guill. Le Conquérant
207	BEN	Le Petit Goulet	76640	FAUVILLE EN CAUX	ES	12,75	07/06/1978	F	0760072R	2	Guill. Le Conquérant
104	BENSAID	25 RUE CARILLON	76430	ANNECY LE VIEUX	L	13,475	16/10/2003	F	0760072R	1	Guill. Le Conquérant
204	JOHN	rue des lupins	76430	LA REMUEE	L	13,475	28/07/1976	F	0760072R	2	Guill. Le Conquérant
107	SAINT MAXENT	10 RUE DU HAVRE	76640	LILLEBONNE	ES	12,75	28/09/1995	F	0760072R	1	Guill. Le Conquérant
105	AVENEL	MARYSE LEBLANC	76600	HARFLEUR	L	12,975	30/09/1994	F	0760023M	1	Jehan Ango
205	COLIN	Rue FOCH	76600	LE HAVRE	L	12,975	09/11/1978	F	0760023M	2	Jehan Ango
209	DAVID	chemin des vallées	76930	OCTEVILLE SUR MER	STT	12,592	05/03/1976	M	0760023M	a	Jehan Ango
...											

**6.5 Exercice :** On veut ordonner les élèves par lycée, pour le même lycée par bac puis par ordre alphabétique des noms.

**7. Calcul (Moyenne, max, min, var) sur une colonne de données.** Même principe que les requêtes de comptage vue en paragraphe 3.

Exemple : La requete suivante afficher nombre etudiants, la moyenne de la promo, la varianc, la note max et la note note\_min

`ELECT count(*) as nombre_etudiants, avg(MOYENNE) AS moy_promo, var(MOYENNE) as varianc_promoe, max(MOYENNE) as note_max, min(MOYENNE) as note_min FROM ETUDIANT ;` **Résultat**

nombre_etudiants	moy_promo	varianc_promoe	note_max	note_min
19	12,6877894736842	0,946826953216414	14,442	10,833

7.1/ Donner la note moyenne par bac : `SELECT ETUDIANT.BAC, avg(MOYENNE) AS moyenne_par_bac FROM ETUDIANT GROUP BY BAC;` **Résultat :**

BAC	moyenne_par_bac
ES	12,75
L	13,225
S	12,6125
STI	11,63
STT	13,517

**7.2 Exercice :** Donner la note moyenne par lycée.

Indication : `SELECT NOM_LYCEE, AVG(MOYENNE) AS MOYENNE_PAR_LYCEE FROM ETUDIANT, LYCEE WHERE LYCEE.ID_LYCEE = ETUDIANT.ID_LYCEE GROUP BY NOM_LYCEE;`