

BASE DE DONNEE

TRAVAUX PRATIQUES N°2

SNOUSSI ANIS
BARGHOUDA MOHAMED LAMINE

REALISE PAR :

SNOUSSI Anis
BARGHOUDA Mohamed
Lamine

COMPTE RENDU

BUT :

- S'habituer à utiliser la Langage SQL pour manipuler directement une base de données



Base de données à atteindre

tp2 etudiant numu : int(11) nom : varchar(10) prenom : varchar(10) # adr : int(11)	tp2 adresse num : int(11) # numr : int(11) nom : varchar(20) code : varchar(11) carac : varchar(1) ville : varchar(20)	tp2 ue code : varchar(11) libelle : varchar(30) # nbh : int(11) # cleens : int(11)
tp2 etudiantue numetu : int(11) numue : varchar(11) # notecc : int(11) # noteexam : int(11)		tp2 enseignant num : int(11) nom : varchar(10) prenom : varchar(10) # age : int(11) ville : varchar(10) # nbheures : int(11)

Etapes de création en SQL :

1. Création du table **Etudiant**

```
CREATE TABLE Etudiant ( Numero INT(8) NOT NULL, Nom VARCHAR(15) NOT NULL, Prenom VARCHAR (15) NOT NULL , PRIMARY KEY (Numero));
```

2. Création du table **UE**

```
CREATE TABLE UE ( Code INT(10) NOT NULL , Libelle VARCHAR(15) NOT NULL, NbHrAssociees INT(3) NOT NULL , PRIMARY KEY (Code));
```

3. Création du table **Adresse**

```
CREATE TABLE Adress ( Numero INT NOT NULL AUTO INCREMENT , Rue VARCHAR(40) NOT NULL , Ville VARCHAR(20) NOT NULL , CodePostal INT(6) NOT NULL, Caractere VARCHAR(1) NOT NULL , PRIMARY KEY (Numero));
```

4. Création du table **Enseignant**

```
CREATE TABLE Enseignant ( Numero INT(8) NOT NULL , Nom VARCHAR(15) NOTNULL , Prenom VARCHAR(15) NOT NULL , Age INT(3) NOT NULL , Ville VARCHAR(40) NOT NULL , NbHrEns INT NOT NULL , PRIMARY KEY (Numero) ;
```

5. Ajout d'une ligne

```
ALTER TABLE Etudiant ADD Adresse VARCHAR (40) AFTER Prenom;  
ALTER TABLE ue ADD enseignant int (8) AFTER NbHrAssociees;
```

6. Ajout d'un lien

```
ALTER TABLE Etudiant ADD CONSTRAINT liaison1 FOREIGN KEY (Adresse) REFERENCES Adresse.Numero;  
ALTER TABLE ue ADD CONSTRAINT liaison2 FOREIGN KEY (enseignant) REFERENCES enseignant.Numero;
```

7. Ajout de données

✓ Ajout dans la table Address

```
INSERT INTO address (NUMRUE,Caractere,Rue,CodePostal,Ville) VALUES (3, 'b', "Jean médecin", "06000", "Nice");
INSERT INTO address (NUMRUE,Caractere,Rue,CodePostal,Ville) VALUES (10, ' ', "Barla", "06000", "Nice");
INSERT INTO address (NUMRUE,Caractere,Rue,CodePostal,Ville) VALUES (10, ' ', "Jean Jaures", "06200", "Cagnes");
```

✓ Ajout dans la table Etudiant

```
INSERT INTO etudiant (Numero,nom,prenom,address) VALUES (1001, "Nom1", "prenom1",1);
INSERT INTO etudiant (Numero,nom,prenom,address) VALUES (1002, "Nom2", "prenom2",2);
INSERT INTO etudiant (Numero,nom,prenom,address) VALUES (1003, "Nom3", "prenom3",3);
INSERT INTO etudiant (Numero,nom,prenom,address) VALUES (1004, "Nom1", "prenom4",4);
```

✓ Ajout dans la table Enseignant

```
INSERT INTO enseignant (Numero,nom,prenom,age,NbHrEns,ville) VALUES (1,"Menez", "Gilles", 25, 35, "Antibes");
INSERT INTO enseignant (Numero,nom,prenom,age,NbHrEns,ville) VALUES (2,"Lahire", "Philippe", 26, 30, "Nice");
INSERT INTO enseignant (Numero,nom,prenom,age,NbHrEns,ville) VALUES (3,"Kounalis", "Emanuel", 27, 28, "Nice");
INSERT INTO enseignant (Numero,nom,prenom,age,NbHrEns,ville) VALUES (4,"Renevier", "Philippe", 21, 28, "Nice");
```

✓ Ajout dans la table ue

```
INSERT INTO ue (code,libelle,NbHrAssociees,enseignant) VALUES ("SL2IBD", "Base de Données", 24, 2);
INSERT INTO ue (code,libelle,NbHrAssociees,enseignant) VALUES ("SL2IPI", "Programmation impérative", 36, 1);
INSERT INTO ue (code,libelle,NbHrAssociees,enseignant) VALUES ("SL2IAL", "Algorithmique", 32, 3);
INSERT INTO ue (code,libelle,NbHrAssociees,enseignant) VALUES ("SL2IPW", "Programmation WEB", 26, 4);
```

8. Création du table EtudiantUE

```
CREATE TABLE EtudiantUE (NoteCC INT(2) NULL DEFAULT NULL , NoteExam INT(2) NULL DEFAULT NULL,etudiant int(8) NOT NULL ,ue varchar(10) NOT NULL, PRIMARY KEY(etudiant,ue));
ALTER TABLE etudiantue ADD CONSTRAINT liaison3 FOREIGN KEY (etudiant) REFERENCES etudiant.Numero;
ALTER TABLE etudiantue ADD CONSTRAINT liaison4 FOREIGN KEY (ue) REFERENCES ue.Code;
```

✓ Ajout dans la table Etudiantue

```
INSERT INTO etudiantue (etudiant,ue,NoteCC,NoteExam) VALUES (1001, "SL2IBD", 10, 11);
INSERT INTO etudiantue (etudiant,ue,NoteCC,NoteExam) VALUES (1001, "SL2IPI", 8, 10);
INSERT INTO etudiantue (etudiant,ue,NoteCC,NoteExam) VALUES (1002, "SL2IBD", 10, 11);
INSERT INTO etudiantue (etudiant,ue,NoteCC,NoteExam) VALUES (1002, "SL2IPI", 8, 10);
INSERT INTO etudiantue (etudiant,ue,NoteCC,NoteExam) VALUES (1003, "SL2IBD", 10, 11);
INSERT INTO etudiantue (etudiant,ue,NoteCC,NoteExam) VALUES (1003, "SL2IPI", 8, 10);
INSERT INTO etudiantue (etudiant,ue,NoteCC,NoteExam) VALUES (1003, "SL2IAL", 12, 13);
```

9. Consultation et affichage des Informations dans la base de données :

✓ Consultation des noms et prénoms de tous les étudiants :

Showing rows 0 - 3 (4 total, Query took 0.0003 seconds.)

```
SELECT nom , prenom FROM etudiant
```

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

nom	prenom
Nom1	prenom1
Nom2	prenom2
Nom3	prenom3
Nom1	prenom4

✓ Consultation des libelles des UEs :

Showing rows 0 - 3 (4 total, Query took 0.0003 seconds.)

```
SELECT libelle , cleens FROM ue
```

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

libelle	cleens
Base de Données	2
Programmation impérative	1
Algorithmique	3
Programmation WEB	4

✓ Consultation des codes uniques et leurs villes correspondantes dans la table adresse :

Showing rows 0 - 1 (2 total, Query took 0.0003 seconds.)

```
SELECT DISTINCT code , ville FROM adresse
```

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

code	ville
O6000	Nice
O6200	Cagnes

✓ Consultation des tous les données concernant les étudiants dont le nom est : « Nom1 »

✓ Showing rows 0 - 1 (2 total, Query took 0.0004 seconds.)

```
SELECT * from etudiant WHERE nom like 'Nom1'
```

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

	numu	nom	prenom	adr
<input type="checkbox"/> Edit Copy Delete	1001	Nom1	prenom1	1
<input type="checkbox"/> Edit Copy Delete	1004	Nom1	prenom4	2

✓ Consultation des numtu des etudiantUE dont le numue est « SL2IBD »

✓ Showing rows 0 - 2 (3 total, Query took 0.0003 seconds.)

```
SELECT numetu FROM etudiantue WHERE numue like 'SL2IBD'
```

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

numetu
1001
1002
1003

✓ Consultation des enseignants dont le prénom est « Philippe » :

✓ Showing rows 0 - 1 (2 total, Query took 0.0003 seconds.)

```
SELECT * FROM enseignant WHERE prenom like 'Philippe'
```

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

	num	nom	prenom	age	ville	nbheures
<input type="checkbox"/> Edit Copy Delete	2	Lahire	Philippe	26	Nice	30
<input type="checkbox"/> Edit Copy Delete	4	Renevier	Philippe	21	Nice	28

☐ Check all
 With selected:
 ☐ Edit
 ☐ Copy
 ☐ Delete
 ☐ Export

✓ Consultation des noms dans table adresse dont les villes sont « Nice » :

✓ Showing rows 0 - 1 (2 total, Query took 0.0003 seconds.)

```
SELECT nom FROM adresse WHERE ville like 'Nice'
```

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

nom
Jean médecin
Barla

✓ Consultation des numetu dans le table etudiantUE dont le note est supérieur à 8 :

✓ Showing rows 0 - 3 (4 total, Query took 0.0003 seconds.)

```
SELECT numetu FROM etudiantue WHERE notecc > 8
```

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

numetu
1001
1002
1003
1003

✓ Consultation de tous les données des enseignant dont le prénom contient « ll » ou « pp » :

✓ Showing rows 0 - 2 (3 total, Query took 0.0004 seconds.)

```
SELECT * FROM enseignant WHERE prenom like '%ll%' OR prenom like '%pp%'
```

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

	num	nom	prenom	age	ville	nbheures
<input type="checkbox"/> Edit Copy Delete	1	Menez	Gilles	25	Antibes	35
<input type="checkbox"/> Edit Copy Delete	2	Lahire	Philippe	26	Nice	30
<input type="checkbox"/> Edit Copy Delete	4	Renavier	Philippe	21	Nice	28

↑ ☐ Check all With selected: Edit Copy Delete Export

10. Suppression et Intégrité

- ✓ Suppression de l'enregistrement dans la table EtudiantUE avec l'id étudiant 1003, et l'UE "SL2IAL"

✓ 0 ligne affectée. (traitement en 0,0000 seconde(s).)

```
DELETE FROM etudiantue WHERE etudiant LIKE 1003 AND ue LIKE 'SL2IAL'
```

- ✓ Suppression des étudiants ayant le nom « Nom1 »

✓ 0 ligne affectée. (traitement en 0,0000 seconde(s).)

```
DELETE FROM etudiant WHERE Nom LIKE 'NOM1'
```

Remarque : On peut effacer une ligne d'une table avec tous ses relations (dans le cas d'un Foreign Key) tout en utilisant « ON DELETE CASCADE » lors du définition du FOREIGN KEY.