Chapitre 05 :Bases de données

Partie 4

T^{ale} NSI 20 septembre 2021

Requêtes SQL

Requête et résultat

Une requête est une commande SQL et renvoie une table. On se replace dans le contexte du chapitre précédent.

Sélection d'attributs

SELECT nom, prenom FROM Auteur;

Sélection d'attributs

SELECT nom, prenom FROM Auteur;

⊞nom	· mprenom ·
1 Ammaniti	Niccolo
2 Avallone	Silvia
3 Camus	Albert
4 Hamilton	Peter
5 Hugo	Victor
6 Murgia	Michela
7 Rhode James	Montague
8 Tolkien	John

Sélection de tous les attributs

SELECT *
FROM Auteur;

Sélection de tous les attributs

```
SELECT *
FROM Auteur;
```

	ആid_auteur:	nom_pays :	™nom :	⊞prenom :	⊞date_naissance :
1	1	. France	Hugo	Victor	1802-02-26
2	2	! France	Camus	Albert	1913-11-07
3	4	Italie	Avallone	Silvia	1948-04-13
4	5	Italie	Ammaniti	Niccolo	1966-09-25
5	6	Italie	Murgia	Michela	1972-06-03
6	7	Royaume-Uni	Hamilton	Peter	1960-03-02
7	8	Royaume-Uni	Tolkien	John	1892-01-03
8	9	Royaume-Uni	Rhode James	Montague	1862-08-01

Sélection avec condition

```
SELECT nom, date_naissance
FROM Auteur
WHERE date(date_naissance) < '1900';</pre>
```

Sélection avec condition

```
SELECT nom, date_naissance
FROM Auteur
WHERE date(date_naissance) < '1900';</pre>
```

	■nom •	■date_naissance :
1	Hugo	1802-02-26
2	Tolkien	1892-01-03
3	Rhode James	1862-08-01

Sélection avec conditions multiples

```
SELECT nom, date_naissance
FROM Auteur
WHERE date(date_naissance) < '1900'
AND nom_pays = 'France';</pre>
```

Sélection avec conditions multiples

```
SELECT nom, date_naissance
FROM Auteur
WHERE date(date_naissance) < '1900'
AND nom_pays = 'France';</pre>
```

```
■ nom · ■ date_naissance · 1 Hugo 1802-02-26
```

Renommer les colonnes

```
SELECT titre AS Titre_ouvrage, num_isbn AS Reference_ISBN
FROM Livre
WHERE date(annee) > '2015';
```

Renommer les colonnes

```
SELECT titre AS Titre_ouvrage, num_isbn AS Reference_ISBN
FROM Livre
WHERE date(annee) > '2015';
```

	⊞Titre_ouvrage	:	⊞Reference_ISBN:
1	les misérables		9782072730672
2	et je t'emmène		9782221133651
3	d'acier		9782867465987
4	salvation		9791093835334

Fonction COUNT

SELECT COUNT(titre) AS Nb_Livres_avant_2015
FROM Livre

Fonction COUNT

```
SELECT COUNT(titre) AS Nb_Livres_avant_2015
FROM Livre
```

```
■ Nb_Livres_avant_2015 · 4
```

Autres fonctions similaires

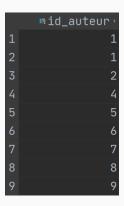
Fonctions MIN, MAX, SUM et AVG (moyenne).

Sans élimination:

```
SELECT id_auteur
FROM Ecrire;
```

Sans élimination:

```
SELECT id_auteur
FROM Ecrire;
```



Avec élimination:

SELECT DISTINCT id_auteur
FROM Ecrire;

Avec élimination:

```
SELECT DISTINCT id_auteur
FROM Ecrire;
```

	জid_auteur∶
1	1
2	2
3	4
	5
5	6
6	7
7	8
8	9

Ordonner les tuples

Ordonner les noms dans l'ordre croissant :

```
SELECT nom, prenom FROM Auteur ORDER BY nom ASC;
```

Ordonner les tuples

Ordonner les noms dans l'ordre croissant :

```
SELECT nom, prenom FROM Auteur ORDER BY nom ASC;
```



Ordonner les tuples

Ordonner les noms dans l'ordre croissant :

```
SELECT nom, prenom FROM Auteur ORDER BY nom ASC;
```



Pour l'ordre décroissant on utilise **DESC** .

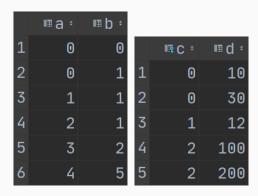
Jointures

Principe

Considérons 2 tables T1 et T2 et supposons que c est une clé étrangère qui fait référence à b.

Principe

Considérons 2 tables T1 et T2 et supposons que c est une clé étrangère qui fait référence à b.



Principe

Considérons 2 tables T1 et T2 et supposons que c est une clé étrangère qui fait référence à b.

	≡ a ÷	⊞b ÷			
1	0	0		⊪ C ÷	≡ d ÷
2	0	1	1	0	10
3	1	1	2	0	30
4	2	1	3	1	12
5	3	2	4	2	100
6	4	5	5	2	200

Voici table qui est la jointure T1 et T2 selon la condition b=c :

	≡ a ÷	⊞þ÷	III C ÷	≡ d ÷
1	0	0	0	10
2	0	0	0	30
3	0	1	1	12
4	1	1	1	12
5	2	1	1	12
6	3	2	2	100
7	3	2	2	200

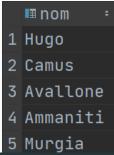
	≡ a ÷	⊞þ ÷	III C ÷	⊞ d ÷
1	0	0	0	10
2	0	0	0	30
3	0	1	1	12
4	1	1	1	12
5	2	1	1	12
6	3	2	2	100
7	3	2	2	200

C'est la table obtenue en faisant correspondre chaque tuple de T1 avec chaque autre tuple de T2 tel que b et c soient égaux.

Produire la table des noms des auteurs venant de pays de plus de 61 millions d'habitants :

Produire la table des noms des auteurs venant de pays de plus de 61 millions d'habitants :

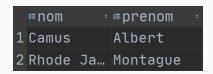
Produire la table des noms des auteurs venant de pays de plus de 61 millions d'habitants :



Produire la table des noms et prénoms des auteurs ayant écrit un livre dont le titre comporte « la » :

Produire la table des noms et prénoms des auteurs ayant écrit un livre dont le titre comporte « la » :

Produire la table des noms et prénoms des auteurs ayant écrit un livre dont le titre comporte « la » :



Mises à jour

Insérer un nouveau tuple dans la table Auteur :

Insérer un nouveau tuple dans la table Auteur :

```
INSERT INTO Auteur VALUES
     (128,'France','Leleu','Frédéric','1974-05-16');
```

Insérer un nouveau tuple dans la table Auteur :

```
INSERT INTO Auteur VALUES
     (128,'France','Leleu','Frédéric','1974-05-16');
```

Les colonnes doivent être dans le même ordre qu'à la création, sinon utiliser

Insérer un nouveau tuple dans la table Auteur :

```
INSERT INTO Auteur VALUES
     (128,'France','Leleu','Frédéric','1974-05-16');
```

Les colonnes doivent être dans le même ordre qu'à la création, sinon utiliser

```
INSERT INTO Auteur VALUES (nom,id_auteur)
    ('Leleu',128);
```

Insérer un nouveau tuple dans la table Auteur :

```
INSERT INTO Auteur VALUES
  (128,'France','Leleu','Frédéric','1974-05-16');
```

Les colonnes doivent être dans le même ordre qu'à la création, sinon utiliser

```
INSERT INTO Auteur VALUES (nom,id_auteur)
    ('Leleu',128);
```

Les colonnes non renseignées prendront par défaut la valeur **NULL** ce qui peut poser problème.

DELETE

Supprimer les tuples de **Ecrire** dont l'auteur a l'id_auteur 1 :

DELETE

Supprimer les tuples de **Ecrire** dont l'auteur a l'id_auteur 1 :

DELETE FROM Ecrire WHERE id_auteur = 1;

DELETE

Supprimer les tuples de **Ecrire** dont l'auteur a l'id_auteur 1 :

```
DELETE FROM Ecrire WHERE id_auteur = 1;
```

Penser aux contraintes de références (clé étrangères) : si on supprime un tuple et qu'un tuple d'une autre table fait référence à celui qu'on supprime, cela provoquera une erreur.

UPDATE

Mettre à jour l'id du tuple de **Auteur** dont le nom est Hugo

UPDATE

Mettre à jour l'id du tuple de **Auteur** dont le nom est Hugo

```
UPDATE Auteur
SET id_auteur = 1024
WHERE nom = 'Hugo';
```

UPDATE

Mettre à jour l'id du tuple de **Auteur** dont le nom est Hugo

```
UPDATE Auteur
SET id_auteur = 1024
WHERE nom = 'Hugo';
```

Penser aux contraintes de références (clé étrangères) lors de la mise à jour.