Structured Query Language

Création : table

Création de table : **CREATE TABLE**, nom de la table, intitulés des attributs avec leur type.

Création : table

Création de table : **CREATE TABLE**, nom de la table, intitulés des attributs avec leur type.

Exemple

CREATE TABLE Célébrités (
Id_Cel **INT**,
Nom **VARCHAR**(255),
Prénom **VARCHAR**(255),
Naissance **YEAR**,
Nationalité **VARCHAR**(255),
);

Création : table

Création de table : **CREATE TABLE**, nom de la table, intitulés des attributs avec leur type.

Exemple

CREATE TABLE Célébrités (
Id_Cel **INT**,
Nom **VARCHAR**(255),
Prénom **VARCHAR**(255),
Naissance **YEAR**,
Nationalité **VARCHAR**(255),
);

Il existe un grand nombre de types (int, varchar (chaînes de caractères), date, etc...), ne pas hésiter à aller lire la documentation !

Et les clés dans tout ça?

Clé Primaire : **PRIMARY KEY**

ID_Cel **INT**,
PRIMARY KEY (ID_Cel)

Et les clés dans tout ça?

Clé Primaire : **PRIMARY KEY**

ID_Cel **INT**,
PRIMARY KEY (ID_Cel)

Clé Étrangère : **FOREIGN KEY REFERENCES**

(dans une autre table)
ID_Cel **INT**,
FOREIGN KEY (ID_Cel) **REFERENCES** Célébrités(ID_Cel)

Insertion

Ajouter une nouvelle donnée dans une table se fait avec les mots-clés **INSERT INTO**, le nom de la table, et le mot-clé **VALUES** suivi des données à insérer.

Insertion

Ajouter une nouvelle donnée dans une table se fait avec les mots-clés **INSERT INTO**, le nom de la table, et le mot-clé **VALUES** suivi des données à insérer.

Exemple

INSERT INTO Célébrités **VALUES** (1, “Lovelace”, “Ada”, 1815, “britannique”);

Insertion

Ajouter une nouvelle donnée dans une table se fait avec les mots-clés **INSERT INTO**, le nom de la table, et le mot-clé **VALUES** suivi des données à insérer.

Exemple

INSERT INTO Célébrités **VALUES** (1, “Lovelace”, “Ada”, 1815, “britannique”);

Attention !!

Les données doivent être du bon type, et entrées dans le même ordre que les attributs de la table !

Exercice

- Rajouter les célébrités suivantes à notre table : George Boole, logiciel britannique né en 1815, et Grace Hopper, informaticienne américaine née en 1906.

Projection/Sélection

Nous allons chercher à accéder aux éléments de la base de données, en les sélectionnant.
Le mot-clé est **SELECT**, suivi des attributs qui nous intéressent, puis de **FROM** et le nom de la table. On peut sélectionner sur les colonnes (**projection**) ou sur les lignes (**sélection**).

Projection/Sélection

Nous allons chercher à accéder aux éléments de la base de données, en les sélectionnant.
Le mot-clé est **SELECT**, suivi des attributs qui nous intéressent, puis de **FROM** et le nom de la table. On peut sélectionner sur les colonnes (**projection**) ou sur les lignes (**sélection**).

Exemple

SELECT Nom **FROM** Célébrités;

Projection/Sélection

Nous allons chercher à accéder aux éléments de la base de données, en les sélectionnant.
Le mot-clé est **SELECT**, suivi des attributs qui nous intéressent, puis de **FROM** et le nom de la table. On peut sélectionner sur les colonnes (**projection**) ou sur les lignes (**sélection**).

Exemple

SELECT Nom **FROM** Célébrités;

Choisir ses colonnes (projection)

- **SELECT** * permet de prendre toutes les colonnes
- Plusieurs colonnes : **SELECT** Col1, Col2, ... **FROM** ...;
- Pas de doublons : **DISTINCT** (**SELECT DISTINCT** Nationalité **FROM** Célébrités;)

Projection/Sélection

Choisir ses lignes : le mot-clé **WHERE** (sélection)

- Le mot-clé **WHERE** permet d'ajouter des conditions sur les données
- Exemple d'utilisation : **SELECT** * **FROM** Célébrités **WHERE** Naissance $>$ 1900;
- Plusieurs conditions avec les mots-clés **AND**, **OR**, et **NOT**

Projection/Sélection

Choisir ses lignes : le mot-clé **WHERE** (sélection)

- Le mot-clé **WHERE** permet d'ajouter des conditions sur les données
- Exemple d'utilisation : **SELECT** * **FROM** Célébrités **WHERE** Naissance $>$ 1900;
- Plusieurs conditions avec les mots-clés **AND**, **OR**, et **NOT**

Ordonner ses données : **ORDER BY**

- Ordre croissant : **SELECT** * **FROM** Célébrités **ORDER BY** Année_Naissance;
- Ordre décroissant : **SELECT** * **FROM** Célébrités **ORDER BY** Année_Naissance **DESC**;

Projection/Sélection

Des fonctions d'agrégation permettent d'effectuer des calculs sur les données. Par exemple : **MIN**, **MAX**, **SUM**, et **COUNT**.

Retourner la date de naissance la plus ancienne

SELECT MIN(Naissance) **FROM** Célébrités

Projection/Sélection

Des fonctions d'agrégation permettent d'effectuer des calculs sur les données. Par exemple : **MIN**, **MAX**, **SUM**, et **COUNT**.

Retourner la date de naissance la plus ancienne

SELECT MIN(Naissance) **FROM** Célébrités

Exercice

- Écrire la requête sélectionnant tous les *Noms* et *prénoms* des célébrités de l'informatique.
- Écrire la requête sélectionnant toutes les *Célébrités* américaines.
- Écrire la requête sélectionnant tous les noms et prénoms des célébrités classées selon leur date de naissance : de la plus actuelle à la plus ancienne.
- Compter le nombre de célébrités présentes dans la base de données.

Nouvelle table, pour la suite

Pour les besoins de démonstration, créons la nouvelle table Inventions :

Inventions

```
CREATE TABLE Inventions (
ID_Inv INTEGER,
Date_Inv YEAR,
Nom VARCHAR(255),
ID_Cel VARCHAR(255),
PRIMARY KEY (ID_Inv),
FOREIGN KEY (ID_Cel) REFERENCES Célébrités(ID_Cel)
);
```

Jointure

Il est possible de recouper des données entre plusieurs tables différentes, au moyen d'une *jointure*.

Jointure

Il est possible de recouper des données entre plusieurs tables différentes, au moyen d'une *jointure*.

Syntaxe d'une jointure : liste des inventions dont les inventeurs sont nés après 1930

SELECT Inventions.Nom **FROM** Inventions **JOIN** Célébrités **ON** Inventions.ID_Cel = Célébrités.ID_Cel **AND** Célébrités.Naissance > 1900

Jointure

Il est possible de recouper des données entre plusieurs tables différentes, au moyen d'une *jointure*.

Syntaxe d'une jointure : liste des inventions dont les inventeurs sont nés après 1930

SELECT Inventions.Nom **FROM** Inventions **JOIN** Célébrités **ON** Inventions.ID_Cel = Célébrités.ID_Cel **AND** Célébrités.Naissance > 1900

Nota Bene

Mettre le nom de la table devant l'attribut (cf POO) est une bonne pratique, nécessaire lorsque le même nom d'attribut est utilisé dans les différentes tables.

Jointure

Il est possible de recouper des données entre plusieurs tables différentes, au moyen d'une *jointure*.

Syntaxe d'une jointure : liste des inventions dont les inventeurs sont nés après 1930

SELECT Inventions.Nom **FROM** Inventions **JOIN** Célébrités **ON** Inventions.ID_Cel = Célébrités.ID_Cel **AND** Célébrités.Naissance > 1900

Nota Bene

Mettre le nom de la table devant l'attribut (cf POO) est une bonne pratique, nécessaire lorsque le même nom d'attribut est utilisé dans les différentes tables.

Autres types de jointures

D'autres jointures sont possibles, mais dépassent le cadre du programme. Celle-ci est une jointure interne (**INNER JOIN**).

Jointure

Il est possible d'utiliser des **alias** qui rendent l'écriture des jointures plus facile.

Jointure

Il est possible d'utiliser des **alias** qui rendent l'écriture des jointures plus facile.

Syntaxe d'une jointure : liste des inventions dont les inventeurs sont nés après 1930

SELECT Inventions.Nom **FROM** Inventions as I **JOIN** Célébrités as C **ON** I.ID_Cel = C.ID_Cel **AND** C.Naissance > 1900

Exercice

- Écrire la requête renvoyant les noms des célébrités ayant créé une invention avant 1900 ainsi que le nom de leur invention.
- Écrire la requête renvoyant la date des inventions des célébrités britanniques.

Mise à jour/suppression

Mettre à jour une donnée : **UPDATE SET**

UPDATE Célébrités **SET** Prenom = “Alan Mathison ” **WHERE** Nom = “Turing”;

Mise à jour/suppression

Mettre à jour une donnée : **UPDATE SET**

UPDATE Célébrités **SET** Prenom = “Alan Mathison ” **WHERE** Nom = “Turing”;

Supprimer une donnée : **DELETE**

DELETE FROM Célébrités **WHERE** Prénom = “George”;

Exercice

- La recherche est souvent plus un travail d'équipe qu'individuel. On se propose de le préciser en changeant le nom de l'invention "Déchiffrement d'Enigma" en "Déchiffrement d'Enigma (avec son équipe)".
- Supprimer la célébrité dont la clé primaire est 4.