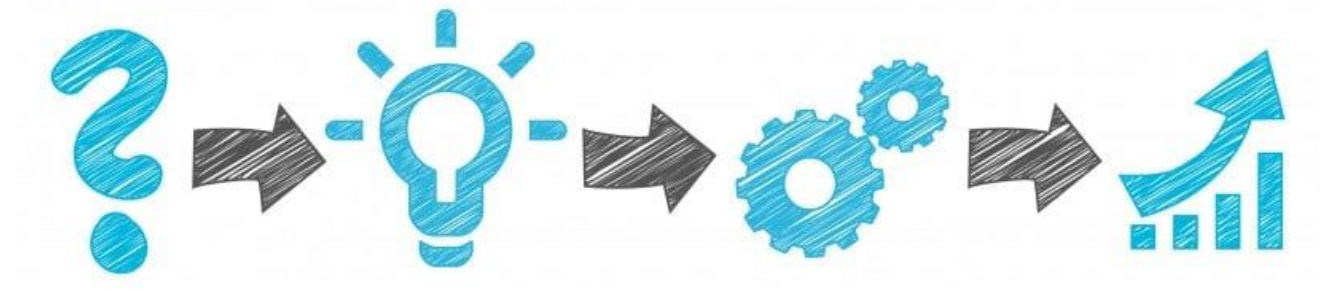


Création et gestion d'une base de données

Le Langage de Définition des Données



Sommaire

- Le LDD : Langage de Définition des Données
- Normalisation des noms, les bonnes pratiques
- Les types de données
- Les contraintes
- Création d'une table

Le langage de définition des données (LDD)

Ce sont les requêtes SQL qui permette de gérer la structure d'une base.

- Commande de trois types :
 - La création : **CREATE**
 - La modification : **ALTER**
 - La suppression : **DROP**
- Ces commandes sont ensuite utilisées avec les objets suivants :
 - Les bases de données : **DATABASE**
 - Les tables : **TABLE**
 - Les vue : **VIEW**

Normalisation des noms, les bonnes pratiques

- Ne pas utiliser de mot clés SQL
- Ne jamais utiliser de caractères spéciaux (espace, accents, ...) dans les nom de bases, tables ou champs.
- Eviter les abréviations
- Utiliser un préfixe dans le nom des champs pour une table données
- Utiliser une convention pour nommer les clés **primaires** et **étrangères** :
 - Le suffixe **ID** pour les clés primaires
 - Le suffixe **FK** pour les clés étrangères

EPREUVES
EP_ID INT(11)
EP_DATE DATE
EP_HEURE TIME
EP_TYPE_FK INT(11)
EP_COMPET_FK INT(11)

Les types de données

Type données du dictionnaire	Type données BDD	Description
Chaîne de caractères	TEXT	Chaîne de caractères alphabétiques
Chaîne de caractères	CHAR(n)	Chaîne alphanumériques ayant une longueur fixe de n caractères
Chaîne de caractères	VARCHAR(n)	Chaîne alphanumériques pouvant contenir aux maximum n caractères
Numérique	SMALLINT	Nom entier court
Numérique	INT(n)	Nombre entier long
Numérique	NUMERIC(n, d)	Nombre de n chiffres maximum et d décimales
Numérique	FLOAT	Nombre réel à virgule flottante
Date	DATE	Date au format AAAA-MM-JJ
Date	TIME	Heure au format HH:MM:SS
Date	DATETIME	Date-heure au format AAAA-MM-JJ HH:MM:SS

Les contraintes

Une contrainte permet d'affiner les valeurs accepté dans un champ donné.

- **(NOT) NULL** : Indique si un champ peut contenir la valeur NULL ou pas.
- **DEFAULT <valeur>** : Assigne une valeur par défaut à un champ.
- **UNIQUE** : Indique qu'un champ ne peut pas contenir de doublons.
- **CHECK (<expression>)** : ajoute un teste de validation des données d'u champ.
- Contrainte d'intégrité fonctionnelles :
 - **PRIMARY KEYG** : définit une clés primaire.
 - **FOREIGN KEY** : définit une clés étrangère.

Création d'une table

```
CREATE TABLE CLUB(  
    CL_ID INT NOT NULL PRIMARY KEY,  
    CL_NOM VARCHAR(30) NOT NULL  
);
```

```
CREATE TABLE COUREUR(  
    CO_ID INT NOT NULL PRIMARY KEY,  
    CO_PRENOM VARCHAR(30) NOT NULL,  
    CO_NOM VARCHAR(30) NOT NULL,  
    CO_NAUSSANCE DATE NOT NULL,  
    CO_SEXE TINYINT(1) NOT NULL CHECK(CO_SEXE IN (1, 2)),  
    CO_CLUB_FK INT NOT NULL,  
    FOREIGN KEY (CO_CLUB_FK) REFERENCES CLUB(CL_ID)  
);
```

```
INSERT INTO CLUB (CL_ID, CL_NOM) VALUES  
(1, 'Club de Cannes'),  
(5, 'Un autre club de Cannes'),  
(13, 'Club de Marseille');
```

```
INSERT INTO COUREUR (CO_ID, CO_PRENOM, CO_NOM, CO_NAUSSANCE, CO_SEXE, CO_CLUB_FK) VALUES  
(1, 'Bruce', 'Willice', '2001-06-01', 1, 1),  
(2, 'Nicol', 'Patrick', '1998-01-06', 2, 13),  
(3, 'Zelda', 'Princesse', '1988-12-25', 2, 5);
```

```
SELECT * FROM COUREUR;
```