

ACTIVITES BASE DE DONNEES

ACTIVITE 1 :

1. Créer les 4 tables suivantes :

La table "client1" de schéma relationnel :

S1 = ((numclient, INTEGER (clé primaire - autovaleur = OUI)), (nomclient, CHAR (15 caractères - saisie requise = OUI)), (prenom, CHAR (15 caractères - saisie requise = OUI)), (age, INTEGER (saisie requise = OUI)), (sexe, CHAR (1 caractère)))

La table "client2" de schéma relationnel : S2 = S1

La table "produit" de schéma relationnel :

S3 = ((numproduit, INTEGER (clé primaire - autovaleur = OUI)), (nomproduit, CHAR (20 caractères - saisie requise = OUI)), (prix, NUMERIC (longueur = 5, décimale = 2 - saisie requise = OUI)), (stock, BOOL (saisie requise = OUI)))

La table "achat" de schéma relationnel :

S4 = ((numachat, INTEGER (clé primaire - autovaleur = OUI)), (numclient, INTEGER (saisie requise = OUI)), (numproduit, INTEGER (saisie requise = OUI)), (date, DATE (saisie requise = OUI)), (promotion, BOOL (saisie requise = OUI)))

2. Remplir les tables "client1", "client2" et "produit" avec les données suivantes en utilisant les requêtes Sql.

numclient	nomclient	prenom	age	sexe
0	Hachard	Samuel	34	H
1	Berthy	Anissa	23	F
2	Allard	Kevin	45	H
3	Courcelle	Aline	67	F
4	Kim	Sung-Ah	45	F
5	Ben chaid	Cherif	32	H
6	Keita	Salif	13	H
7	Dilan	Bob	78	H
8	Toto	Salif	13	H

"client1"

numclient	nomclient	prenom	age	sexe
0	Gomez	Antonio	47	H
1	Garland	Todd	14	H
2	James	Tania	30	F
3	Leer	Amanda	72	F
4	Pichon	Laure	24	F
5	Lin	Yu-Ting	27	F
6	Berthy	Anissa	23	F
7	Allard	Kevin	45	H

"client2"

numproduit	nomproduit	prix	stock
0	savon	2,30	<input checked="" type="checkbox"/>
1	dentifrice	1,70	<input checked="" type="checkbox"/>
2	shampooing	3,20	<input type="checkbox"/>
3	rasoir	7,50	<input checked="" type="checkbox"/>
4	mouchoir	1,60	<input checked="" type="checkbox"/>
5	mousse	3,40	<input type="checkbox"/>
6	parfum	10,30	<input checked="" type="checkbox"/>

"produit"

3. Sélectionner le contenu de la table « produit » avec la commande Sql appropriée.
4. Sélectionner l'enregistrement numproduit 4 de la table « produit » avec la commande Sql appropriée.
5. Sélectionner les clients de la table « client 2 » agés de plus de 30 ans avec la commande Sql appropriée.
6. Trier les clients de la table « client 1 » par âge croissant avec la commande Sql appropriée.

ACTIVITE 2 :

Le service Web de collecte enregistre les données des stations de mesures dans la base de données MySQL « vigicrues ». Une partie du schéma conceptuel de la base de données est disponible dans la documentation PPB.

Il faut ajouter une table « dataNiveau » pour stocker les valeurs de niveau d'eau. Cette table respecte le modèle des autres capteurs et contient un champ supplémentaire « Debit » pour le calcul du débit d'eau.

Q16. Préciser, à l'aide des documentations PP8 et PP9, le code SQL permettant de créer la table **dataNiveau**.

Q17. Préciser, à l'aide des documentations PP8 et PP9, le code SQL permettant d'insérer un nouvel enregistrement dans la table **dataNiveau** avec les valeurs suivantes :

IDstation = 1, niveau instantané = 0,17 m, niveau moyen = 0,16 m, débit instantané = 0,6 m³/s.

Suite à l'exécution de la requête SQL suivante :

```
DELETE FROM station WHERE IDstation = 1
```

le serveur MySQL a répondu :

```
#1451 - Cannot delete or update a parent row
```

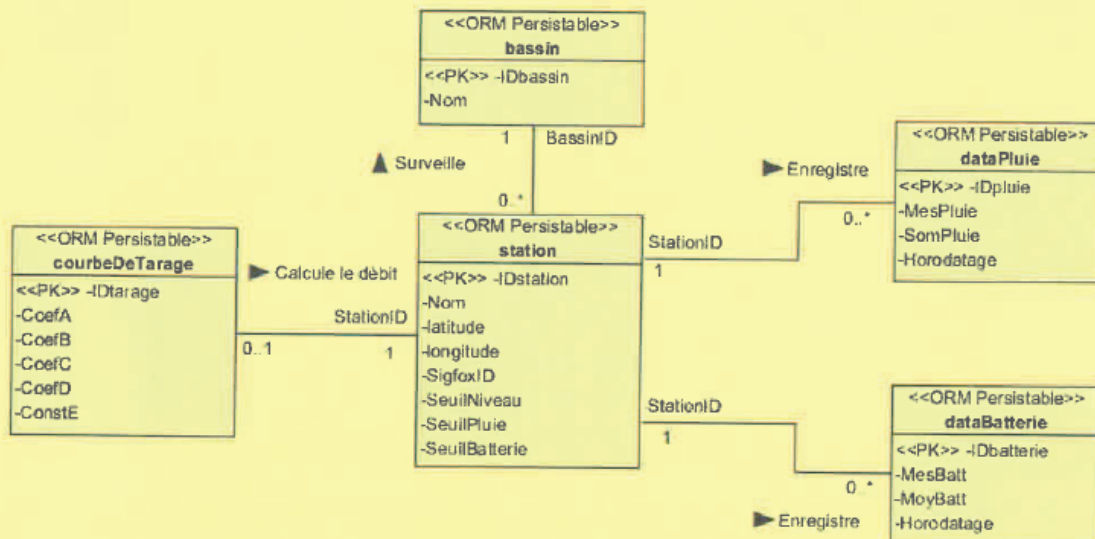
Q18. Expliquer, à l'aide des documentations PP8 et PP9, la raison de cette erreur et proposer une solution au problème. Les requêtes SQL ne sont pas demandées.

Le site Web de consultation utilise une base de données identique à la base « vigicrues ».

Q19. Préciser, à l'aide des documentations PP8 et PP9, le code SQL permettant de sélectionner toutes les moyennes de niveaux d'eau de la station qui s'appelle « Wimille » en commençant par la plus récente.

Documentation PP8 : Base de données « vigicrues »

Le service web de collecte enregistre les données des stations de mesures dans la base de données MySQL « vigicrues ». Cette base de données est décrite par le schéma conceptuel suivant (notation UML2) :



Ce modèle conceptuel génèrera le code SQL suivant pour créer la table dataBatterie :

```
CREATE TABLE databatterie (
    IDbatterie int(10) NOT NULL AUTO_INCREMENT,
    StationID int(10) NOT NULL,
    MesBatt float DEFAULT NULL,
    MoyBatt float DEFAULT NULL,
    Horodatage timestamp NULL DEFAULT CURRENT_TIMESTAMP,
    PRIMARY KEY (IDbatterie),
    CONSTRAINT FOREIGN KEY (StationID) REFERENCES station
(IDstation)
);
```

Le champ Horodatage des tables est rempli automatiquement par Mysql avec la date courante lors de chaque insertion d'un nouvel enregistrement.

La table courbeDeTarage contient les coefficients de la fonction polynomiale d'ordre 4 qui interpole la courbe de tarage établie par l'hydrologue.

Débit = CoefA * Niveau⁴ + CoefB * Niveau³ + CoefC * Niveau² + CoefD * Niveau + ConstE

Documentation PP9 : Rappel des syntaxes SQL de Mysql

Utiliser (rendre active) une base de données existante :	USE nom_de_la_base ;
Créer une base de données :	CREATE DATABASE nom_de_la_base ;
Supprimer une base de données	DROP DATABASE nom_de_la_base ;
Créer une table dans la base de données active:	CREATE TABLE nomTable (id INT NOT NULL AUTO_INCREMENT, champ1 DOUBLE, champ2 VARCHAR, champ3 TIMESTAMP NOT NULL, ..., PRIMARY KEY(id)) ;
Sélectionner toutes les informations de la table :	SELECT * FROM nomTable ;
Sélectionner seulement les informations d'un champ :	SELECT nomChamp FROM nomTable ;
Sélectionner tous les champs de la table nomTable correspondant à deux critères.	SELECT * FROM nomTable WHERE nomChamp1 = 'poste' AND nomChamp3 < 12 ;
Sélectionner sur plusieurs tables nomTable1.nomChamp1 est clé primaire. nomTable2.nomChamp4 est une clé étrangère vers nomTable1.	SELECT * FROM nomTable1, nomTable2 WHERE nom_table1.nomChamp1 = nom_table2.nomChamp4 ;
Trier les résultats par ordre croissant	SELECT nomChamp FROM nomTable ORDER BY nomChamp ASC ;
Trier les résultats par ordre décroissant	SELECT nomChamp FROM nomTable ORDER BY nomChamp DESC ;
Écrire une nouvelle entrée dans une table de BDD :	INSERT INTO nomTable(champ1, champ2) VALUES('valeur_texte1', 0.867) ;
Modifier les informations de l'entrée dont le champ id = 51 :	UPDATE nomTable SET nomChamp1=10, valeur2=32 WHERE id=51 ;
Supprimer une entrée dans une table	DELETE FROM nomTable WHERE champ1='valeur1' ;

ACTIVITE 3 :

Les questions suivantes concernent la base de données et son évolution.

Consulter les documentations PP4 et PP5

La base de données « solaire » est constituée de deux tables.

La première sert à l'authentification des utilisateurs.

La seconde permet de stocker les différentes mesures.

Q1. Expliquer le rôle de la clé primaire « idMesure » de la table « mesures ».

Une acquisition a donné pour résultats : uPan = 24.0, iPan = 18.8, energie = 321.2, uBat = 23.8, temperature = 34.0.

Le champ horodatage est affecté automatiquement lors de l'enregistrement dans la base de données.

Q2. Écrire en langage SQL la requête permettant d'écrire cette nouvelle acquisition dans la base de données.

Pour une version plus évoluée du système, il est envisagé de pouvoir avoir plusieurs contrôleurs MPPT.

Il est donc proposé d'ajouter une table contenant les informations concernant chaque MPPT.

<i>MPPT</i>
<i><u>IdMPPT</u></i>
<i>CAN_ID</i>
<i>nom</i>

Le champ « idMPPT » correspond à la clé primaire de notre nouvelle table nommée « MPPT ».

Le champ « CAN_ID » correspond à l'identifiant associé à chaque MPPT sur le bus CAN.

Le champ « nom » correspond au nom donné pour la carte.

Q3. À l'aide des **documentations PP4 et PP5**, donner la requête SQL permettant de créer la table MPPT dans base de données nommée « solaire ».

Q4. Quel champ doit-on ajouter à la table « mesures » afin de modéliser la relation entre les mesures et la carte MPPT les ayant effectuées.

Q5. **Dans le document réponses**, compléter le schéma de la base de données en indiquant le nouveau champ de la table « mesures » ainsi que les cardinalités entre la table « mesures » et la table « MPPT ».

Q6. Donner le rôle du champ que vous avez ajouté à la table « mesures ».

Écrire en langage SQL la requête permettant de récupérer les valeurs de l'énergie liée au MPPT dont le champ « CAN_ID » a pour valeur 3.

Documentation PP4 : Schéma relationnel de la base de données

La base de données est composée d'une table « mesures » et de quatre utilisateurs présents dans la table « utilisateurs ».

mesures	utilisateurs
<u>idMesure</u> horodatage tension_pan courant_pan energie tension_bat temperature	<u>idUser</u> login mdp

Documentation PP5 : Rappel des syntaxes SQL de Mysql

Utiliser (rendre active) une base de données existante :	USE nom_de_la_base;
Créer une base de données :	CREATE DATABASE nom_de_la_base;
Supprimer une base de données	DROP DATABASE nom_de_la_base;
Créer une table dans la base de données active:	CREATE TABLE nomTable (id INT NOT NULL AUTO_INCREMENT, champ1 DOUBLE, champ2 VARCHAR, champ3 TIMESTAMP NOT NULL, ..., PRIMARY KEY(id)) ;
Lister la structure d'une table :	DESCRIBE nomTable;
Sélectionner toutes les informations de la table :	SELECT * FROM nomTable ;
Sélectionner seulement les informations d'un champ :	SELECT nomChamp FROM nomTable ;
Sélectionner tous les champs de la table nomTable correspondant à deux critères.	SELECT * FROM nomTable WHERE nomChamp1 = 'poste' AND nomChamp3 < 12 ;
Sélectionner sur plusieurs tables nomTable1.nomChamp1 est clé primaire. nomTable2.nomChamp4 est une clé étrangère vers nomTable1.	SELECT * FROM nomTable1, nomTable2 WHERE nom_table1.nomChamp1 = nom_table2.nomChamp4 ;
Écrire une nouvelle entrée dans une table de BDD :	INSERT INTO nomTable(champ1, champ2) VALUES('valeur1', 'valeur2') ;
Modifier les informations de l'entrée dont le champ id = 51 :	UPDATE nomTable SET nomChamp1=10, valeur2=32 WHERE id=51 ;

DOCUMENT REPONSE

Q5. Compléter le schéma de la base de données en indiquant le nouveau champ de la table « mesures » ainsi que les cardinalités entre la table « mesures » et la table « MPPT ».

MPPT
<u>idMPPT</u> CAN_ID nom

mesures
<u>idMesure</u> horodatage tension_pan courant_pan energie tension_bat temperature

utilisateurs
<u>idUser</u> login mdp