

Master CNAM - ESAIP

UE :

Introduction à la gestion des données à large échelle

Plateforme Web pour la gestion des données des concours de dessins

Document Réponses

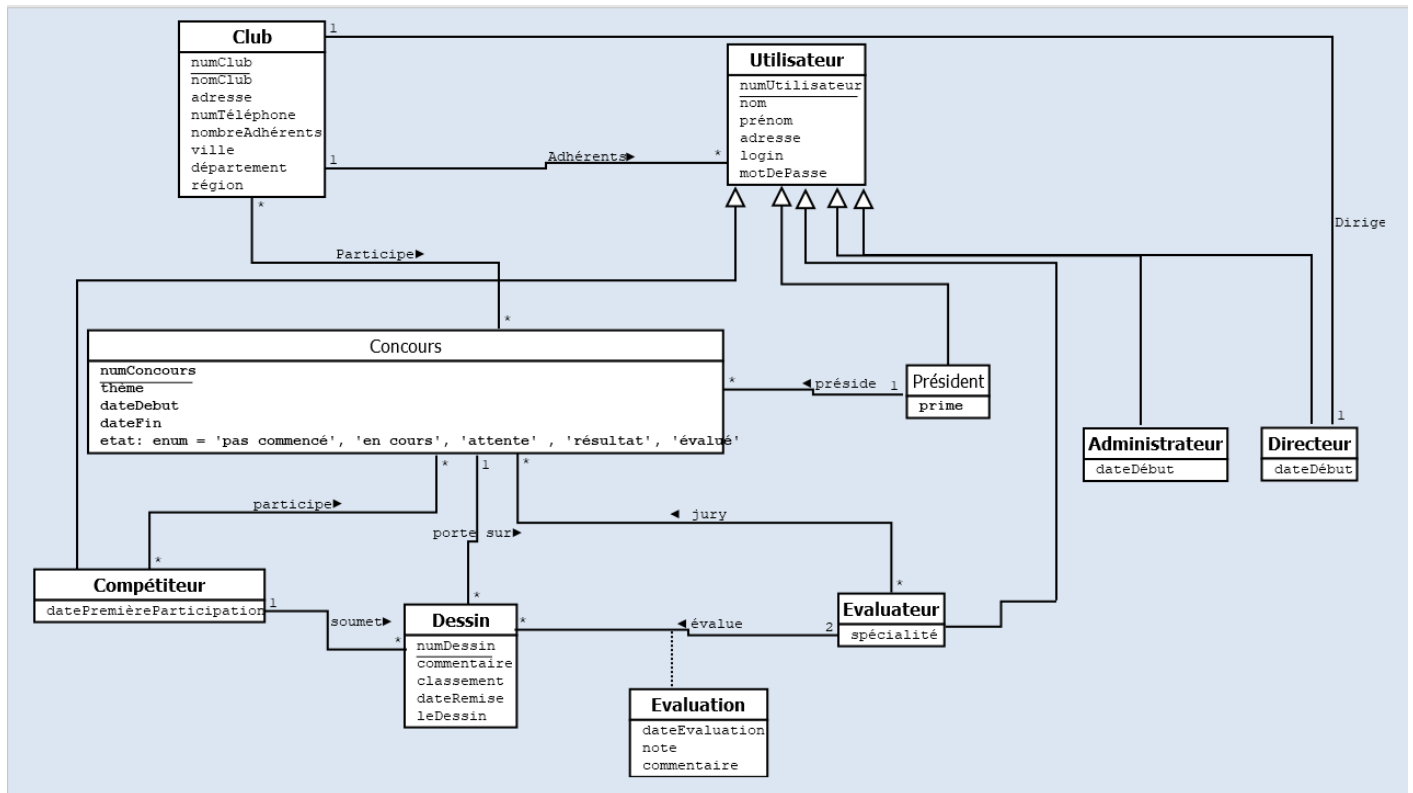
Ce document réponse doit être mis dans l'archive à déposer sur le campus à une date ultérieure qui vous sera confirmée. Une version PDF de ce document devra être déposée. Veuillez inscrire sur le tableau ci-dessous vos noms et prénoms (3 par projet). Utiliser les noms des membres de l'équipe pour nommer votre Archive. Vous devez lire attentivement le document « CahierDesCharges » disponible sur le campus avant de répondre aux questions posées.

Nom	Prénom
POTTIER	Bastian
YATAN	EMIRJAN

Question 1 : Le Schéma conceptuel de la base de données

Après avoir lu le cahier des charges, proposé un schéma conceptuel en UML. Veuillez utiliser un éditeur graphique UML pour réaliser vos schémas (*DIA, LucidChart, ArgoUML, ...*). Une fois le schéma réalisé, insérer-le ci-dessous. Veuillez compléter le schéma conceptuel qui a été discuté en cours en proposant des attributs supplémentaires dans au moins 3 classes.

Schéma conceptuel UML



Question 2 : Analyse du schéma conceptuel

Après avoir lu le cahier des charges, vous allez aborder une phase d'analyse concernant les contraintes. En effet, le système d'information que vous devez concevoir doit respecter un ensemble de contraintes que nous pouvons formuler comme suit :

- Il ne peut y avoir que 4 concours par année durant les 4 saisons (1 concours par saison).
- Un évaluateur ne pourra pas évaluer plus de 8 dessins en tout dans un même concours,
- Un dessin doit être évalué par deux évaluateurs qui constituent un Jury.
- Tout président d'un concours ne pourra pas être évaluateur ou compétiteur de ce même concours.
- Tout compétiteur ne peut pas déposer plus de trois dessins dans un même concours donné
- Un concours pour être organisé doit mobiliser au moins 6 clubs.
- Tout club qui participe à un concours donné doit mobiliser au moins 6 compétiteurs et 3 évaluateurs.
- Un évaluateur d'un concours ne peut pas concourir dans celui-ci

Pour chacune de ces contraintes, préciser lesquelles sont explicitement définies dans le schéma conceptuel proposé précédemment. Pourquoi ? Quelles sont celles qui ne sont pas représentées dans le schéma ?

Lors du remplissage de votre base de données (commandes insert) vous allez devoir respecter toutes ces contraintes afin d'avoir une base de données cohérente par rapport au domaine d'application.

Contrainte 1

Non. La contrainte sur le nombre de concours par année n'est pas représentée dans le schéma. Il n'existe aucune règle ou mécanisme dans le modèle conceptuel qui limite la création de concours à 4 par an. Ce genre de contrainte est de type métier et nécessite une validation applicative.

Contrainte 2

Non. Le schéma ne limite pas le nombre d'évaluations par évaluateur dans un concours. Le lien entre Evaluation et Evalueur ne contient aucune restriction sur le nombre maximum d'évaluations qu'un évaluateur peut faire. Cela devrait être géré au niveau de la logique métier dans l'application.

Contrainte 3

Non. Le schéma permet plusieurs évaluateurs par dessin mais n'impose pas exactement deux. Le lien entre Evaluation, Dessins, et Evalueur est de type plusieurs-à-plusieurs, ce qui autorise 0, 1, 2 ou plus d'évaluateurs pour un dessin sans contrôle. Une vérification de "exactement 2" doit être faite via des règles métiers ou des triggers.

Contrainte 4

Non. Rien n'empêche un président d'être aussi évaluateur ou compétiteur dans le même concours.

Le modèle n'interdit pas à un utilisateur d'avoir plusieurs rôles simultanément dans un concours, car les rôles (President, Competiteur, Evalueur) sont tous liés à la table Utilisateurs sans contrainte croisée.

Contrainte 5

Non. Le schéma ne limite pas le nombre de dessins qu'un compétiteur peut soumettre à un concours.

Le lien entre Dessins et Concours/Competiteurs ne contient aucune contrainte sur le nombre de dessins par concours. Cette règle n'est pas implémentée dans le schéma.

Contrainte 6

Non. Le schéma ne vérifie pas que 6 clubs minimum participent à un concours.

Le lien ParticipeClub permet de lier un ou plusieurs clubs à un concours, mais ne fixe pas de minimum. Cela nécessite une vérification applicative ou une contrainte via trigger.

Contrainte 7

Non. Le schéma ne garantit pas qu'un club mobilise au moins 6 compétiteurs et 3 évaluateurs. Il n'y a pas de lien direct ou de règle dans le schéma qui lie la participation d'un club au nombre de compétiteurs et d'évaluateurs qu'il fournit. Cela doit être vérifié au niveau de l'application ou de la logique métier.

Contrainte 8

Non. Le schéma n'empêche pas un évaluateur d'être aussi compétiteur dans un même concours.

Comme pour la contrainte 4, les rôles sont gérés séparément dans des tables différentes mais référencent tous Utilisateurs, sans contrainte croisée. Rien n'interdit à un même utilisateur d'être dans plusieurs rôles pour un même concours.

Question 3 : Le schéma logique

En suivant les règles de passage entre un schéma conceptuel vers un schéma logique de bases de données relationnelles, proposez ci-dessous le schéma logique correspondant au schéma conceptuel. Veuillez proposer la spécification textuelle puis graphique du schéma logique. Pour la spécification graphique, vous pouvez utiliser des logiciels comme DIA, PlantUML¹, ou tout autre logiciel permettant de dessiner un schéma logique de BD relationnel. Les clés étrangères et les clés primaires doivent être bien désignées dans le formalisme textuel.

a) Formalisme textuel pour le schéma logique

Formalisme Textuel

Club (numClub, numDirecteur*, nomClub, adresse, numTéléphone, nombreAdhérents, ville, département, région)

Concours (numConcours, numPrésident*, thème, dateDebut, dateFin, état)

Utilisateurs (numUtilisateur, nom, prénom, adresse, login, motDePasse, numClub*)

Dessin (numDessins, commentaire, classement, dateRemise, leDessin, numConcours*, numCompétiteur*)

Compétiteurs (numCompétiteur*, datePremièreParticipation)

Evaluation (numDessins*, numEvaluateur*, dateEvaluation, note, commentaire)

Evaluateur (numEvaluateur*, spécialité)

Président (numPrésident*, prime)

Directeur (numDirecteur*, dateDébut)

Administrateur (numAdministrateur*, dateDébut)

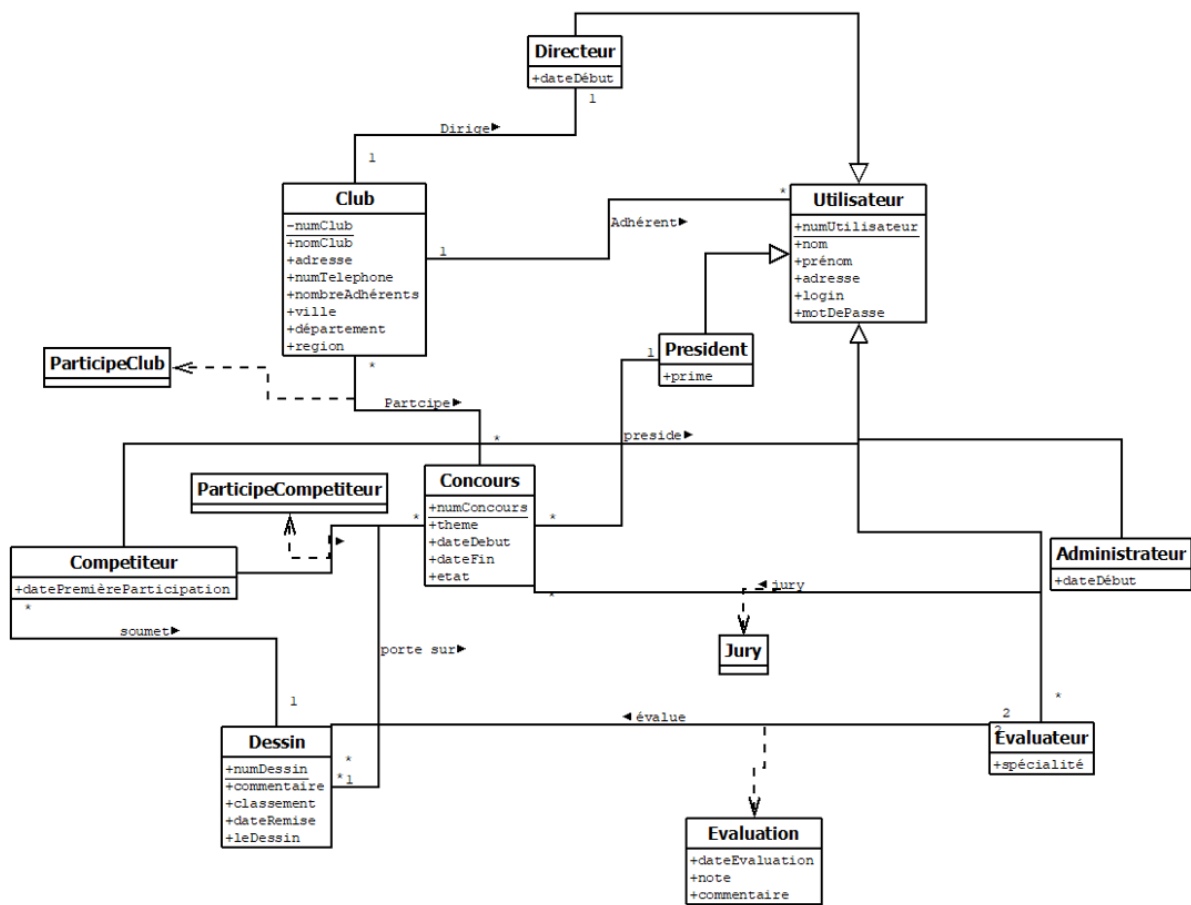
Jury (numEvaluateur*, numConcours*)

Participe Compétiteur (numCompétiteur*, numConcours*)

Participe Club (numClub*, numConcours*)

¹ <http://fr.plantuml.com/download.html> (vous pouvez télécharger le .jar à cette adresse)

b) Formalisme graphique pour le schéma logique



Question 4 : Le schéma physique

À partir du schéma logique que vous avez proposé, déduire le schéma physique conforme à un SGBD MYSQL/PHPMYADMIN. Les types des différents attributs doivent être judicieusement choisis. Vous devez exécuter votre schéma physique sous MYSQL/PHPMYADMIN, et une fois validé, vous devez sauvegarder votre fichier ***creationConcoursDessins.sql*** que vous déposerez avec l'archive complète du projet. Veuillez recopier le schéma physique ci-dessous.

```
-- Accorder tous les privilèges à l'utilisateur esaip
GRANT ALL PRIVILEGES ON *.* TO 'esaip'@'%' IDENTIFIED BY 'esaip' WITH GRANT OPTION;
FLUSH PRIVILEGES;
```

```
CREATE DATABASE ConcoursDessin;
USE ConcoursDessin;
```

```
CREATE TABLE Club (
numClub INT AUTO_INCREMENT PRIMARY KEY,
numDirecteur INT NOT NULL,
nomClub VARCHAR(255) NOT NULL,
adresse VARCHAR(255),
numTelephone VARCHAR(15),
nombreAdherents SMALLINT,
ville VARCHAR(100),
departement VARCHAR(100),
region VARCHAR(100)
);
```

```
CREATE TABLE Utilisateurs (
numUtilisateur INT AUTO_INCREMENT PRIMARY KEY,
nom VARCHAR(100) NOT NULL,
prenom VARCHAR(100) NOT NULL,
adresse VARCHAR(255) NOT NULL,
login VARCHAR(50) UNIQUE NOT NULL,
motDePasse VARCHAR(255) NOT NULL,
numClub INT NOT NULL,
FOREIGN KEY (numClub) REFERENCES Club(numClub)
);
```

```
CREATE TABLE Directeur (
numDirecteur INT PRIMARY KEY,
dateDebut DATE NOT NULL,
FOREIGN KEY (numDirecteur) REFERENCES Utilisateurs(numUtilisateur)
);
```

```

CREATE TABLE Concours (
numConcours INT AUTO_INCREMENT PRIMARY KEY,
numPresident INT NOT NULL,
theme VARCHAR(255) NOT NULL,
dateDebut DATE NOT NULL,
dateFin DATE NOT NULL,
etat ENUM('non commence', 'en cours', 'en attente des resultats', 'evalue') NOT NULL,
FOREIGN KEY (numPresident) REFERENCES Utilisateurs(numUtilisateur)
);

CREATE TABLE President (
numPresident INT PRIMARY KEY,
prime INT NOT NULL,
FOREIGN KEY (numPresident) REFERENCES Utilisateurs(numUtilisateur)
);

CREATE TABLE Competiteurs (
numCompetiteur INT PRIMARY KEY,
datePremiereParticipation DATE NOT NULL,
FOREIGN KEY (numCompetiteur) REFERENCES Utilisateurs(numUtilisateur)
);

CREATE TABLE Dessins (
numDessins INT AUTO_INCREMENT PRIMARY KEY,
commentaire TEXT,
classement INT,
dateRemise DATE,
leDessin TEXT,
numCINSERT INTO ParticipeClub (numClub,numConcours) VALUES (2,8);

```


Question 5 : Remplissage de la BD

Le schéma physique que vous avez défini et créé sous MYSQL/PHPMYADMIN doit être maintenant rempli par des insertions de tuples dans les différentes tables. Ces insertions devront respecter toutes les contraintes du cahier des charges et suivront donc les règles ci-dessous :

La base de données doit contenir **au moins² les données suivantes** :

- 8 concours, réalisés durant 2 années successives comme 2023 et 2024 (4 concours/année).

Les données à insérer dans la base de données doivent respecter les contraintes du cahier des charges, en particulier les contraintes suivantes :

- Il ne peut y avoir que 4 concours par année durant les 4 saisons (1 concours par saison).
- Un évaluateur ne pourra pas évaluer plus de 8 dessins en tout dans un même concours,
- Un dessin doit être évalué par deux évaluateurs qui constituent un Jury.
- Tout président d'un concours ne pourra pas être évaluateur ou compétiteur de ce même concours.
- Tout compétiteur ne peut pas déposer plus de trois dessins dans un même concours donné
- Un concours pour être organisé doit mobiliser au moins 6 Clubs.
- Tout club qui participe à un concours donné doit mobiliser au moins 6 compétiteurs et 3 évaluateurs. A vous de bien remplir la BD afin d'avoir plusieurs scénarios.
- Un évaluateur d'un concours ne peut pas concourir dans celui-ci

Concernant les évaluations, il serait judicieux de donner des notes différentes pour permettre d'établir un classement. La base de données doit refléter tous les scénarios possibles décrits dans le cahier des charges ; En particulier, la base de données doit contenir des concours avec les différents états proposés : *non commencé, en cours, en attente, évalué*. Il est clair que seuls les concours avec l'état « évalué » seront pertinents pour l'expression des requêtes et l'exploitation de la BD et ils devront donc être très majoritaire dans la BD.

Un fichier ***insertionConcoursDessins.sql*** avec toutes les requêtes d'insertions dans la BD doit être mis dans l'archive à déposer.

² Dans la perspective de proposer des requêtes SQL et faire une interface en PHP, il est souhaitable d'avoir une BD avec le plus de données possibles. Vous avez donc la liberté de remplir la BD par des données cohérentes.

Question 6 : Interrogation de la Bases de Données (Requêtes SQL)

Proposez 10 requêtes SQL permettant de répondre aux attentes présentées dans le cahier de charges ou que vous jugeriez utiles à l'exploitation du système. Chaque requête sera exprimée en français puis en langage SQL. Parmi les 10 requêtes, les 5 suivantes sont imposées :

Requête 1

Afficher le nom et l'adresse et l'âge de tous les compétiteurs qui ont participé dans un concours en 2024. Vous afficherez aussi la description du concours la date de début et la date de fin. Le club du compétiteur, le département et la région.

```
SELECT
    u.nom AS nomCompetiteur,
    u.prenom AS prenomCompetiteur,
    u.adresse,
    YEAR(CURDATE()) - YEAR(comp.datePremiereParticipation) AS ageCompetiteur,
    c.theme AS descriptionConcours,
    c.dateDebut,
    c.dateFin,
    cl.nomClub AS nomClub,
    cl.departement,
    cl.region
FROM
    Utilisateurs u
JOIN
    Competiteurs comp ON u.numUtilisateur = comp.numCompetiteur
JOIN
    ParticipeCompetiteur pc ON comp.numCompetiteur = pc.numCompetiteur
JOIN
    Concours c ON pc.numConcours = c.numConcours
JOIN
    Club cl ON u.numClub = cl.numClub
WHERE
    YEAR(c.dateDebut) = 2024
LIMIT 0, 25;
```

Requête 2

Afficher par ordre croissant de la note tous les dessins qui ont été évalués en 2023. Vous afficherez les informations suivantes : le numéro du dessin et la note attribuée, le nom du compétiteur, la description du concours et le thème du concours.

```
SELECT
    d.numDessins AS numeroDessin,
    e.note,
    u.nom AS nomCompetiteur,
    c.theme AS themeConcours,
    c.theme AS descriptionConcours
FROM
    Dessins d
JOIN
    Evaluation e ON d.numDessins = e.numDessins
JOIN
    Utilisateurs u ON d.numCompetiteur = u.numUtilisateur

    Concours c ON d.numConcours = c.numConcours
WHERE
    YEAR(e.dateEvaluation) = 2023
ORDER BY
    e.note ASC;
```

Requête 3

Pour cette requête on vous demande d'afficher des informations sur tous les dessins qui ont été évalués et qui sont stockés dans la base. Voici les informations qu'on souhaite voir affichés : le numéro, l'année, la description du concours dans lequel le dessin a été évalué ; le nom du compétiteur ayant proposé le dessin ; le numéro et le commentaire du dessin fait par le compétiteur ; la note et le commentaire de l'évaluation ; le nom de l'évaluateur.

```
SELECT
    d.numDessins AS numeroDessin,
    YEAR(c.dateDebut) AS anneeConcours,
    c.theme AS descriptionConcours,
    u.nom AS nomCompetiteur,
    d.numDessins AS numeroDessin,
    d.commentaire AS commentaireDessin,
    e.note,
    e.commentaire AS commentaireEvaluation,
    eval.nom AS nomEvalueur
FROM
    Dessins d
JOIN
    Evaluation e ON d.numDessins = e.numDessins
JOIN
    Utilisateurs u ON d.numCompetiteur = u.numUtilisateur
```

```

    Utilisateurs eval ON e.numEvalueur = eval.numUtilisateur
JOIN
    Concours c ON d.numConcours = c.numConcours;

```

Requête 4

Nom, prénom et âge des compétiteurs qui ont participé à tous les concours qui ont été organisés en 2023 et 2024. L’affichage doit se faire dans l’ordre croissant des âges.

```

SELECT
    u.nom AS nomCompetiteur,
    u.prenom AS prenomCompetiteur,
    YEAR(CURDATE()) - YEAR(comp.datePremiereParticipation) AS ageCompetiteur --
    Corriger la référence à datePremiereParticipation
FROM
    Utilisateurs u
JOIN
    Competiteurs comp ON u.numUtilisateur = comp.numCompetiteur
JOIN
    ParticipeCompetiteur pc ON comp.numCompetiteur = pc.numCompetiteur -- Corriger
    le nom de la table
JOIN
    Concours c ON pc.numConcours = c.numConcours
WHERE
    c.dateDebut BETWEEN '2023-01-01' AND '2024-12-31'
GROUP BY
    comp.numCompetiteur
HAVING
    COUNT(DISTINCT c.numConcours) = (
        SELECT
            COUNT(*)
        FROM
            Concours
        WHERE
            YEAR(dateDebut) IN (2023, 2024)
    )
ORDER BY
    ageCompetiteur ASC;

```

Requête 5

Nom de la région qui a la meilleure moyenne des notes des dessins proposés. Afficher le nom de la région et la moyenne des notes de cette région.

```
SELECT
    cl.region,
    AVG(e.note) AS moyenneNotes
FROM
    Club cl
JOIN
    Utilisateurs u ON cl.numClub = u.numClub
JOIN
    Competiteurs comp ON u.numUtilisateur = comp.numCompetiteur
JOIN
    Dessins d ON comp.numCompetiteur = d.numCompetiteur
JOIN
    Evaluation e ON d.numDessins = e.numDessins
GROUP BY
    cl.region
ORDER BY
    moyenneNotes DESC
LIMIT 1;
```

Requête 6

Afficher les concours qui n'ont encore reçu aucune évaluation.

```
SELECT c.numConcours, c.theme, c.dateDebut, c.dateFin
FROM Concours c
LEFT JOIN Dessins d ON c.numConcours = d.numConcours
LEFT JOIN Evaluation e ON d.numDessins = e.numDessins
WHERE e.numDessins IS NULL;
```

Requête 7

Afficher pour chaque concours le nombre de dessins soumis.

```
SELECT c.numConcours, c.theme, COUNT(d.numDessins) AS nombreDessins
FROM Concours c
LEFT JOIN Dessins d ON c.numConcours = d.numConcours
GROUP BY c.numConcours, c.theme;
```

Requête 8

Afficher la liste des évaluateurs ainsi que le nombre de dessins qu'ils ont évalués.

```
SELECT u.nom, u.prenom, COUNT(e.numDessins) AS nbEvaluations
FROM Evaluation e
```

```
JOIN Evalueur ev ON e.numEvalueur = ev.numEvalueur
JOIN Utilisateurs u ON ev.numEvalueur = u.numUtilisateur
GROUP BY u.nom, u.prenom;
```

Requête 9

Afficher les compétiteurs ayant reçu au moins une note supérieure ou égale à 18.

```
SELECT DISTINCT u.nom, u.prenom, e.note
FROM Evaluation e
JOIN Dessins d ON e.numDessins = d.numDessins
JOIN Utilisateurs u ON d.numCompetiteur = u.numUtilisateur
WHERE e.note >= 18;
```

Requête 10

Lister les concours avec le nom du président organisateur et le nombre d'évaluateurs affectés.

```
SELECT c.numConcours, c.theme, u.nom AS nomPresident, u.prenom AS prenomPresident,
COUNT(j.numEvalueur) AS nbEvalueurs
FROM Concours c
JOIN Utilisateurs u ON c.numPresident = u.numUtilisateur
LEFT JOIN Jury j ON c.numConcours = j.numConcours
GROUP BY c.numConcours, c.theme, u.nom, u.prenom;
```

Question 7 : Une interface WEB pour le Système d'information

Pour l'interface, nous avons mis à disposition une interface python directement executable dans le projet pour effectuer les requêtes. (Nous sommes des très mauvais dev)

Exécution de la requête 1...

nomCompetiteur	prenomCompetiteur	adresse	ageCompetiteur	descriptionConcours	dateDebut	dateFin	nomClub	departement	region
CompNom301	CompPren301	Adresse Comp301	2	Thème 2024-1	2024-03-20	2024-03-27	Club A	Paris	Ile-de-France
CompNom302	CompPren302	Adresse Comp302	2	Thème 2024-1	2024-03-20	2024-03-27	Club A	Paris	Ile-de-France
CompNom303	CompPren303	Adresse Comp303	2	Thème 2024-1	2024-03-20	2024-03-27	Club A	Paris	Ile-de-France
CompNom304	CompPren304	Adresse Comp304	2	Thème 2024-1	2024-03-20	2024-03-27	Club A	Paris	Ile-de-France
CompNom305	CompPren305	Adresse Comp305	2	Thème 2024-1	2024-03-20	2024-03-27	Club A	Paris	Ile-de-France
CompNom306	CompPren306	Adresse Comp306	2	Thème 2024-1	2024-03-20	2024-03-27	Club A	Paris	Ile-de-France
CompNom307	CompPren307	Adresse Comp307	2	Thème 2024-1	2024-03-20	2024-03-27	Club B	Rhône	Auvergne-Rhône-Alpes
CompNom308	CompPren308	Adresse Comp308	2	Thème 2024-1	2024-03-20	2024-03-27	Club B	Rhône	Auvergne-Rhône-Alpes
CompNom309	CompPren309	Adresse Comp309	2	Thème 2024-1	2024-03-20	2024-03-27	Club B	Rhône	Auvergne-Rhône-Alpes
CompNom310	CompPren310	Adresse Comp310	2	Thème 2024-1	2024-03-20	2024-03-27	Club B	Rhône	Auvergne-Rhône-Alpes
CompNom311	CompPren311	Adresse Comp311	2	Thème 2024-1	2024-03-20	2024-03-27	Club B	Rhône	Auvergne-Rhône-Alpes
CompNom312	CompPren312	Adresse Comp312	2	Thème 2024-1	2024-03-20	2024-03-27	Club B	Rhône	Auvergne-Rhône-Alpes
CompNom313	CompPren313	Adresse Comp313	2	Thème 2024-1	2024-03-20	2024-03-27	Club C	Bouches-du-Rhône	Provence-Alpes-Côte d'Azur
CompNom314	CompPren314	Adresse Comp314	2	Thème 2024-1	2024-03-20	2024-03-27	Club C	Bouches-du-Rhône	Provence-Alpes-Côte d'Azur
CompNom315	CompPren315	Adresse Comp315	2	Thème 2024-1	2024-03-20	2024-03-27	Club C	Bouches-du-Rhône	Provence-Alpes-Côte d'Azur
CompNom316	CompPren316	Adresse Comp316	2	Thème 2024-1	2024-03-20	2024-03-27	Club C	Bouches-du-Rhône	Provence-Alpes-Côte d'Azur
CompNom317	CompPren317	Adresse Comp317	2	Thème 2024-1	2024-03-20	2024-03-27	Club C	Bouches-du-Rhône	Provence-Alpes-Côte d'Azur
CompNom318	CompPren318	Adresse Comp318	2	Thème 2024-1	2024-03-20	2024-03-27	Club C	Bouches-du-Rhône	Provence-Alpes-Côte d'Azur
CompNom319	CompPren319	Adresse Comp319	2	Thème 2024-1	2024-03-20	2024-03-27	Club D	Haute-Garonne	Occitanie
CompNom320	CompPren320	Adresse Comp320	2	Thème 2024-1	2024-03-20	2024-03-27	Club D	Haute-Garonne	Occitanie
CompNom321	CompPren321	Adresse Comp321	2	Thème 2024-1	2024-03-20	2024-03-27	Club D	Haute-Garonne	Occitanie
CompNom322	CompPren322	Adresse Comp322	2	Thème 2024-1	2024-03-20	2024-03-27	Club D	Haute-Garonne	Occitanie
CompNom323	CompPren323	Adresse Comp323	2	Thème 2024-1	2024-03-20	2024-03-27	Club D	Haute-Garonne	Occitanie
CompNom324	CompPren324	Adresse Comp324	2	Thème 2024-1	2024-03-20	2024-03-27	Club D	Haute-Garonne	Occitanie
CompNom325	CompPren325	Adresse Comp325	2	Thème 2024-1	2024-03-20	2024-03-27	Club E	Alpes-Maritimes	Provence-Alpes-Côte d'Azur