

Bases de données Projet

I) Modalités

Le projet doit être réalisé par binôme.

La pré-soutenance aura lieu le mercredi 13 et le jeudi 14 novembre lors du TP. Pour l'ordre de passage, vous devrez vous inscrire par binôme sur un lien qui sera indiqué sur Moodle. Lors de la pré-soutenance, vous devrez apporter sur papier (cf. détails dans la suite du sujet) :

- votre schéma entités/associations,
- un schéma relationnel de vos tables,
- une brève description des requêtes personnelles que vous prévoyez pour la partie e) du II).

Ne commencez pas à coder avant la pré-soutenance car il est possible que vous soyez amenés à faire des changements à votre modélisation.

La soutenance devrait avoir lieu en janvier (à confirmer selon les contraintes d'emploi du temps). Vous devrez vous inscrire à un créneau par binôme comme pour la pré-soutenance. Lors de la soutenance, vous ferez une présentation de 15 minutes au cours de laquelle vous présenterez votre modélisation et les fonctionnalités de votre base. Vous devez préparer cette soutenance et elle doit tenir dans le délai imparti. (La présentation se déroule sur machine, il n'est pas nécessaire de préparer des transparents.) Une bonne présentation inclut une démonstration des requêtes avec des jeux de données préparés à l'avance. Pensez à structurer votre soutenance avec un ou plusieurs scripts (cf. annexes).

Vous devez rendre au format électronique sur Moodle la veille de la soutenance avant 18h :

- votre modèle entités/associations,
- un schéma de vos tables,
- le script de création de vos tables,
- le script utilisé pour peupler vos tables,
- le script de vos ordres de gestion,
- le script de vos requêtes.

Chaque requête doit être précédée d'un commentaire qui explique brièvement ce qu'elle fait, et elle doit être accompagnée de jeux de données qui permettent d'en illustrer le bon fonctionnement.

Pour la soutenance, préparez un script interactif et soyez prêts à répondre à des questions concernant le fonctionnement de vos requêtes, les contraintes d'intégrité, les choix de modélisation, les points forts et les points faibles du travail réalisé, les parties incomplètes, les améliorations qui restent à faire.

II) Description du sujet

Vous êtes un agent d'artistes (musiciens, acteurs, etc.) et voulez créer une base de données pour gérer tous vos clients.

a) Spécificités de votre choix

Vous gérez donc un certain type d'artistes (soyez innovants). Ces artistes ont un statut particulier, ils ne sont pas salariés : ils sont rémunérés à chacune de leurs prestations, et vous prélevez une commission. Dans votre base, vous aurez sans doute besoin de ce genre d'informations habituelles (liste non exhaustive) :

- type de contrat (commission, etc.) ;
- engagements des artistes (date, lieu, type, etc.) ;
- rémunérations, droits d'auteur, etc. ;
- emploi du temps de chaque artiste ;
- lieux (salles de concert...) et conditions ;
- etc.

Chaque artiste possède un compte et doit pouvoir se connecter afin de modifier des données le concernant.

Prenez le temps de faire une liste de toutes les informations dont vous avez besoin.

b) Modélisation

Cette partie est à rendre lors de la pré-soutenance les 13-14 novembre.

Donnez un schéma entités/associations pour votre base. Pensez à éviter les redondances dans les informations, tout en conservant toutes les informations pertinentes et nécessaires. Pensez à identifier les contraintes d'intégrité.

En déduire le schéma relationnel (liste des tables) de votre base.

À rendre aussi les 13-14 novembre : un bref descriptif des requêtes SQL personnelles (voir la partie e) du II)).

c) Création et peuplement des tables

Après accord de l'enseignant au moment de la pré-soutenance, vous devrez

- créer les tables provenant de votre schéma relationnel ;
- les peupler par des données suffisamment pertinentes pour les différentes requêtes ci-dessous.

d) Gestion des opérations courantes

Vous devez préparer des ordres SQL qui permettent de gérer les profils des clients :

- inscription d'un nouvel artiste, mise à jour de ses données personnelles, changement de contrat, suppression de son compte, etc. ;
- gestion des différents lieux de production (ajout d'une salle, mise à jour des conditions, etc.).

Référez-vous à l'annexe pour une façon de présenter les ordres de gestion de façon interactive.

e) Autres requêtes

Vous devez également préparer au moins 10 requêtes SQL complexes, accompagnées de données qui permettent de bien les tester. Pensez bien à vos jeux de données, ils seront

notés aussi. Attention, il vaut mieux peu de données bien choisies, que trop de données qui noient les résultats.

Les requêtes imposées

Vous devez donner des requêtes SQL pour les items suivants (une requête pour chaque item).

1. Prochain jour libre pour un artiste particulier.
2. Liste des salles proposant pour tel artiste un cachet supérieur à 1000 euros.
3. Rentabilité d'un artiste particulier sur le dernier mois.
4. Temps de travail d'un artiste particulier sur l'année écoulée.
5. Cachet moyen d'un artiste particulier.
6. Chiffre d'affaire généré au cours du mois dernier par tous les artistes.

Les requêtes personnelles

Vous devez proposer au moins 4 requêtes supplémentaires, par exemple :

- des requêtes présentant des statistiques sur les artistes ;
- des requêtes identifiant des anomalies dans les données (il faut justifier que ces anomalies ne peuvent pas être gérées par des contraintes sur la base) ;
- etc.

La notation tiendra compte de la complexité des requêtes. Pensez à des requêtes demandant la jointure de plusieurs tables, l'utilisation de fonctions d'agrégation (MIN, MAX, SUM...) avec GROUP BY... HAVING, de sous-requêtes, etc.

Annexe

Pour rendre votre projet plus interactif, vous pouvez préparer des requêtes dont certains paramètres seront fournis par l'utilisateur. Par exemple, si vous faites une recherche par ville, vous pourriez avoir une requête comme suit :

```
SELECT nom
  FROM restaurant
 WHERE ville = 'Cancale';
```

Mais comment faire si vous souhaitez faire une requête dont la ville n'est pas connue d'avance ?

Préparation de requêtes paramétrées Vous pouvez créer une requête paramétrée en utilisant la syntaxe suivante.

```
PREPARE recherche_par_ville(VARCHAR) as
SELECT nom
  FROM restaurant
 WHERE ville = $1;
```

Vous pouvez ajouter autant de paramètres que vous souhaitez. Dans la requête, vous utiliserez \$1, \$2... pour en obtenir la valeur. La requête est préparée mais elle n'est pas exécutée. Pour l'exécuter, on tape sous psql :

```
EXECUTE recherche_par_ville('Cancale');
```

La requête est exécutée avec la valeur 'Cancale' à la place du paramètre \$1.

Variables d'environnement et prompt La commande `prompt` de psql permet de faire la saisie au clavier. La valeur est sauvegardée dans une variable d'environnement, `v_ville` dans l'exemple qui suit.

```
\prompt 'Tapez le nom de ville -> ' v_ville
SELECT nom
  FROM restaurant
 WHERE ville = :v_ville;
```

Lorsque la requête est exécutée, `:v_ville` prend la valeur de la variable d'environnement `v_ville`. (NB : il faut taper les apostrophes lors de la saisie sans quoi la chaîne sera interprétée comme un attribut dans la requête.)