

Projet de conception BDD

Un lieu culturel

Conception de bases de données relationnelles

P. Blarre - Université Grenoble Alpes - MIASHS

Consignes Générales	2
Énoncé	3
Partie 1 - Conception (MCD) - 10pts	4
Partie 2 - Production (MLD) - 10pts	5
Partie 3 - Requêtes - 10pts	5
Partie 4 - Contraintes et Triggers - 10pts	6
Barème et consignes	6
En option	7

Consignes Générales

Ce projet est à réaliser en trinôme ou en binôme. Sauf exception validée avec moi, des points seront retirés si vous n'êtes pas en groupe. Utilisez le Discord pour trouver des personnes si besoin.

Merci d'indiquer les noms de votre groupe ici:

https://docs.google.com/spreadsheets/d/1I53Gupikb4_6gGbjMTf6kRbWgUtKITkPRVfWzmFED3c/edit?usp=sharing

Le projet est à rendre sur Moodle avant le 24 Mars 2021 Minuit (24/03/2021 23h59)
La pénalité de retard est de 30% de la note en moins par jour de retard.

Vous aurez besoin d'un ordinateur avec MySQL et MySQL Workbench installés. Lisez le sujet en entier plusieurs fois avant de commencer.

Vous devez soumettre une archive zip contenant:

- Un rapport PDF contenant votre numéro de groupe ainsi que vos noms et filières, votre conception, diagrammes, résultats, etc. ainsi qu'un paragraphe court sur les difficultés rencontrées et comment vous avez travaillé en groupe (outils utilisés, qui a fait quoi, etc.)
- Un fichier *database_create.sql* contenant le code SQL de la création des tables de la base de données ainsi que des vues et des procédures.
- Un fichier *database_data.sql* contenant le code SQL des insertions de données uniquement
- Un fichier *database_queries.sql* contenant le code des requêtes SQL. Veuillez ajouter un commentaire avec le numéro et l'énoncé de la requête avant chaque requête.
- Un fichier *database_triggers.sql* contenant le code de vos triggers ainsi que des requêtes permettant de tester les différentes contraintes et triggers de votre base.

L'archive ZIP que vous m'envoyez doit être nommée de la sorte (remplacez "Groupe1" avec votre numéro de groupe et NOM1, NOM2 et NOM3 avec vos noms de famille) :

Groupe1_NOM1_NOM2_NOM3.zip

L'archive est à déposer sur Moodle.

Si vous n'arrivez pas à déposer votre travail sur Moodle, merci de me contacter avant la date de rendu avec votre travail. Tout retard non justifié sera pénalisé de 30% par jour de retard.

Énoncé

On veut informatiser la gestion d'un lieu culturel en créant une base de données relationnelle. Après un entretien avec les gestionnaires de l'association du lieu culturel, voici les informations dont on dispose :

Notre espace culturel est composé de 3 salles de spectacle de différentes capacités (une de 300 personnes, une de 100 personnes et une de 50 personnes). Des événements sont organisés dans ces 3 salles, parfois dans le cadre d'événements communs, type festival, et parfois en utilisant une ou deux salles uniquement. Des événements et spectacles différents peuvent avoir lieu au même moment s'ils n'utilisent pas les mêmes salles.

Nous accueillons différents types de spectacles : musique, théâtre, performances, lectures, etc. Chaque événement peut proposer un ou plusieurs spectacle. Chaque artiste reçoit un cachet prédéfini pour sa performance, et chaque spectacle a un ou plusieurs tarifs pour le public (tarif plein, tarif réduit, tarif bénévole, tarif jeune, etc.). Un artiste reçoit son paiement après sa performance, il faut savoir si un artiste a été payé ou non. Les artistes sont représentés à minima par leurs nom et description. On veut conserver un historique des événements, spectacles et artistes qui se sont produits dans notre lieu.

Chaque spectacle a un nombre de places limité en fonction de la salle dans laquelle il se déroule. Les personnes du public peuvent réserver des tickets en avance si elles ont un compte dans la base de données. Les tickets réservés pourront être payés à l'avance ou sur place. Si une personne ayant réservé un ticket ne se présente pas 15 minutes au plus tard avant le début du spectacle, alors on considère que son ticket peut être revendu sur place (Les places disponibles ne sont donc plus nécessairement les mêmes 15min avant le début du spectacle). Un événement peut-être annulé. Les tickets peuvent alors être remboursés, mais on veut conserver un historique de ceux-ci.

De nombreux bénévoles participent à la vie du lieu culturel. Ils peuvent être bénévoles "génériques" ou bien avoir de spécificités comme aide à la comptabilité, connaissances techniques musicales, aide à la sécurité etc. Ils ont des tarifs préférentiels sur les événements

et les consommations. Ils peuvent avoir des missions longues (sur l'année, par exemple pour choisir la programmation), et ils peuvent aussi s'inscrire comme bénévole pour les événements, pour préciser qu'ils sont disponibles pour aider.

Certains bénévoles peuvent être désignés responsables d'un pôle (programmation, organisation, sécurité, nettoyage, etc.) et aussi d'un événement. Lors d'événements, on pourra leur assigner une liste de bénévoles pour qu'ils puissent gérer les équipes des autres bénévoles participant à un événement.

Nous avons aussi une dizaine de salariés qui occupent différents postes dans l'association (technicien, administratif, sécurité, etc.) en plus des membres de la présidence de l'association. Certains membres et salariés sont responsables les uns des autres. Les salariés ont un salaire mensuel prédéfini.

On va considérer que toute personne peut être identifiée à minima par ses nom, prénom, date de naissance, adresse, numéro de téléphone et adresse email.

Vous devez proposer une étude qui réponde au mieux à cet énoncé. Les gestionnaires de l'association ne sont pas des informaticiens et ne pourront pas vous aider pour la conception de la base de données.

Vous devez prendre des décisions et justifier vos choix dans votre rapport pour proposer une solution des plus logique, permettant de répondre aux différentes contraintes et requêtes demandées par la suite.

Partie 1 - Conception (MCD) - 10pts

- Extraire les propositions pertinentes de l'énoncé.
- Proposer, sous forme de textes courts, les entités, attributs, associations et cardinalités
- Identifier les dépendances fonctionnelles
- Réaliser un schéma entité-association à l'aide d'un logiciel de votre choix (Diagrams.net, ERDPlus, Lucidchart, Dia ou autre)
- Effectuer une passe de normalisation pour vérifier le schéma (justifier que toutes vos types d'entités et associations sont bien en 3FN)

Votre compte rendu doit contenir les propositions, les dépendances fonctionnelles, et une impression d'écran de votre schéma, ainsi que des justifications de vos choix et des annotations.

Partie 2 - Production (MLD) - 10pts

- Transformer votre schéma E-A en schéma de table sur papier ou dans un logiciel, puis en modèle de base de données dans le logiciel MySQL WorkBench
- Générer le code SQL du diagramme et tester la création de la base de données dans MySQL. Sauvegarder dans un fichier **database_create.sql**.
- Ajouter des données à la base de données
Vous pouvez ajouter des données manuellement ou bien utiliser l'outil filldb: <http://filldb.info/> pour générer des fausses données dans votre bases de données et récupérer le fichier de création de la base avec la fausse données. N'essayez pas forcément de tout générer avec filldb. Essayez de créer quelques données types qui vous permettrons de valider vos requêtes.
- Créer la base de données dans MySQL à l'aide de MySQL WorkBench
- N'oubliez pas d'ajouter des contraintes sur les différentes tables et colonnes (contraintes d'intégrité, contraintes statiques (CHECK), dynamiques et de cascades)

Votre rendu doit contenir une impression d'écran du votre schéma de table et diagramme de classe, les fichiers SQL généré par MySQL WorkBench et un fichier d'importation de vos données

Partie 3 - Requêtes - 10pts

Rédiger l'algèbre relationnel lorsque cela est possible, puis le SQL sous forme de requêtes ou de vues pour chacune des questions suivantes :

1. Liste des bénévoles et de leur age, ainsi que les événements auxquels ils ont participés
2. Montant des prestations réglées aux artistes entre deux dates données
3. Afficher le nombre de places disponibles pour un événement
4. Afficher le nombre de minutes, d'heures, de jours, de mois et d'années avant le début d'un événement
5. Afficher les événements par ordre de meilleures ventes, avec les sommes collectées au total d'une part, et le bénéfice réalisé d'autre part (après paiement des artistes)
6. Proposer une requête d'insertion d'un événement composé de 7 spectacles qui utilisent les 3 salles. Elle peut être composée de plusieurs requêtes et ne doit pas imposer d'identifiant (== elle doit pouvoir fonctionner sans tenir compte des identifiants des autres événements ou spectacles)

7. Les salaires annuels des salariés pour une année donnée
8. Les ventes d'un événements classées par tarifs (tarifs plein, jeune, etc.)
9. Les pourcentages de types de spectacle pour une année donnée
10. Les événements annulés ainsi que le montant des tickets remboursés

*L'algèbre relationnel pourra être écrit à la main puis scanné ou photographié pour être ajouté au compte rendu. Les requêtes SQL doivent être ajoutées au compte rendu, et un fichier **database_queries.sql** doit contenir l'ensemble de vos requêtes pour que je puisse facilement les tester sur votre base de données. Vous pouvez aussi stocker vos requêtes sous forme de vues.*

Partie 4 - Contraintes et Triggers - 10pts

En utilisant des contraintes statiques ou dynamiques, des procédures ou des triggers, proposez des solutions pour vérifier les contraintes suivantes :

1. Un événement peut-être annulé, mais il ne peut pas être supprimé de la base de données, et un ticket vendu peut-être remboursé, mais pas supprimé
2. Vérifier qu'un événement ou un spectacle ne peut pas être créé si un autre événement ou spectacle utilise déjà une des salles sur le même créneau
3. Un bénévole ne peut-être assigné qu'à un seul responsable par événement
4. On ne peut pas vendre plus de tickets que disponible pour un événement par rapport aux capacités des salles utilisées
5. Les tickets réservés mais non payés doivent redevenir disponibles 15min avant le début d'un événement

Vous devez proposer des requêtes permettant de vérifier que les contraintes fonctionnent, ou bien m'indiquer comment les tester.

Barème et consignes

Le barème est donné à titre indicatif. Vous perdrez des points si vous ne respectez pas les consignes de rendu, pour l'orthographe et la présentation générale.

Pas de rédaction à la main, un PDF rédigé avec Google Doc ou autre pour pouvoir travailler en groupe sur le rapport.

Pensez à bien vérifier la création de votre base de données dans MySQL avant d'effectuer votre rendu. Si vous utilisez Windows, essayer de la créer sur une machine Linux ou MacOS et vice-versa. Précisez la version de MySQL utilisée (5.7 ou 8).

Au final, vous devriez pouvoir "enchaîner" vos fichiers SQL : 1) création de la base 2) insertion des données 3) execution des requêtes 4) ajout et tests des vues, procédures ou triggers.

Je vais corriger en lisant tout d'abord votre rapport, puis je vais créer votre base de données pour tester vos requêtes. Si j'obtiens des erreurs lors de la création de la base de données et que je suis obligé de corriger pour que ça fonctionne, vous serez pénalisé.

Pour rappel, les 3 séances de TP (12h) seront dédiées à apprendre à utiliser WorkBench et MySQL et à faire le projet. Je vous conseille de préparer votre conception et d'utiliser WorkBench et MySQL avant les séances de TP pour que nous puissions vous aider au mieux sur les points bloquants.

En option

Cette partie n'est pas du tout obligatoire et devrait être réalisée uniquement pour les intéressé(e)s qui ont tout terminé en avance.

- Transposer le projet pour qu'il fonctionne dans le SGBD [Postgres](#). Si vous l'avez effectué, vous pouvez ajouter une seconde archive zip dans votre archive de rendu nommée postgres.zip.
- Produire un schéma de votre projet avec [DBeaver](#). Si vous l'avez effectué, ajoutez votre fichier dans votre archive de rendu.