

# Projet de conception BDD

## Agence d'intérim

*Conception de bases de données relationnelles*

*P. Blarre - Université Grenoble Alpes - MIASHS*

<b>Consignes Générales</b>	<b>2</b>
<b>Énoncé</b>	<b>3</b>
<b>Partie 1 - Conception (MCD) - 10pts</b>	<b>4</b>
<b>Partie 2 - Production (MLD) - 10pts</b>	<b>4</b>
<b>Partie 3 - Requêtes SQL - 10pts</b>	<b>6</b>
<b>Partie 4 - Contraintes, procédures (routines) et déclencheurs (triggers) - 10pts</b>	<b>7</b>
<b>Barème et consignes</b>	<b>9</b>
<b>En option</b>	<b>10</b>

# Consignes Générales

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

Merci d'indiquer vos binômes sur ce document :

[https://docs.google.com/spreadsheets/d/1I53Gupikb4\\_6gGbjMTf6kRbWgUtKITkPRVfWzmFED3c/edit?usp=sharing](https://docs.google.com/spreadsheets/d/1I53Gupikb4_6gGbjMTf6kRbWgUtKITkPRVfWzmFED3c/edit?usp=sharing)

Le projet est à rendre sur Moodle avant le 5 Avril 2022 Minuit ( **05/04/2022 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 à travailler

Vous devez déposer sur Moodle 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.) et toute remarque que vous jugez pertinente.
- [Un fichier MySQL Workbench model.mwb](#) contenant votre modèle de base de données, un diagramme, les insertions, vues, procédures, triggers, etc.
- [Un fichier database\\_create.sql](#) contenant le code SQL de la création des tables de la base de données ainsi que des vues, triggers et procédures (pas d'insert dans ce fichier).
- [Un fichier database\\_data.sql](#) contenant le code SQL des insertions de données uniquement (pas de create dans ce document).
- [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 les requêtes permettant de tester vos triggers et les différentes contraintes de votre base. Veuillez ajouter un commentaire avec le numéro et l'énoncé de la contrainte avant chaque requête de tests et d'ajouter des commentaires expliquant votre démarche ou le résultat attendu si nécessaire.

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

*Groupe1\_NOM1\_NOM2.zip*

L'archive est à déposer sur Moodle.

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

## Énoncé

On veut informatiser la gestion d'une agence d'intérim qui permet de trouver des emplois. Voici les informations dont on dispose :

*L'agence emploie des salariés qui ont différents rôles (Directeur, manager, secrétaire, accompagnateur, etc.). L'activité principale de l'agence est de proposer des offres d'emploi et des stages à des candidats et de suivre leur parcours professionnel.*

*Les candidats ont des informations personnelles, une description, des diplômes et niveaux d'études, des expériences professionnelles, des métiers et des compétences. Les métiers, compétences et expériences professionnelles peuvent être ajoutés au fur et à mesure de la carrière du candidat. Un candidat est soit disponible soit indisponible s'il est actuellement employé.*

*Les expériences professionnelles peuvent être des CDI, des CDD, des stages, du bénévolat, etc. Elles ont des dates de début et éventuellement de fin.*

*Les métiers sont, par exemple : Développeur Web, Analyste Data, Charpentier, Soudeur, Professeur, etc.*

*Les compétences sont, par exemple : Logiciels bureautiques, Métallurgie, Java, MySQL, Formation, Dessin Technique, etc.*

*Les salariés de l'agence reçoivent et publient des offres d'emploi proposées par des entreprises. Les offres d'emploi ont un intitulé, un salaire brut mensuel défini par un taux horaire (par exemple 20€ de l'heure et 160 heures par mois), sont rattachées à une entreprise et peuvent être liées à des métiers et des compétences. Elles ont aussi un type de contrat (CDI, CDD, stage, etc.), une date de début et une date de fin si le type de contrat n'est pas un CDI.*

*On devra pouvoir enregistrer les candidatures d'un candidat pour une offre d'emploi. Si le candidat a obtenu le poste, l'offre d'emploi doit changer de statut de "ouverte" à "fermée" avec la date de fermeture et être liée au candidat retenu. Le candidat ne pourra pas être ajouté à l'offre d'emploi s'il n'est pas disponible ou si la candidature a déjà été obtenue.*

*Pour chaque offre d'emploi fermée, l'agence touchera une commission de 10% du salaire brut mensuel de l'annonce. Le salarié qui a géré l'offre d'emploi touchera une prime de 10% de la commission touchée par l'agence.*

*On va considérer que toute personne, salarié ou candidat, peut être identifiée à minima par ses nom, prénom, date de naissance, adresse, numéro de téléphone et adresse email. Les salariés peuvent aussi être des candidats et vice versa.*

Vous devez proposer une étude qui réponde au mieux à cet énoncé. Les gestionnaires de l'agence 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, en 3NF, permettant de répondre aux différentes contraintes et requêtes demandées par la suite.

## Partie 1 - Conception (MCD) - 10pts

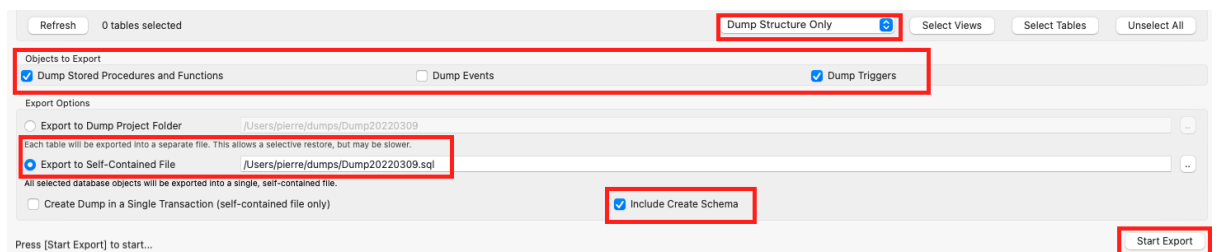
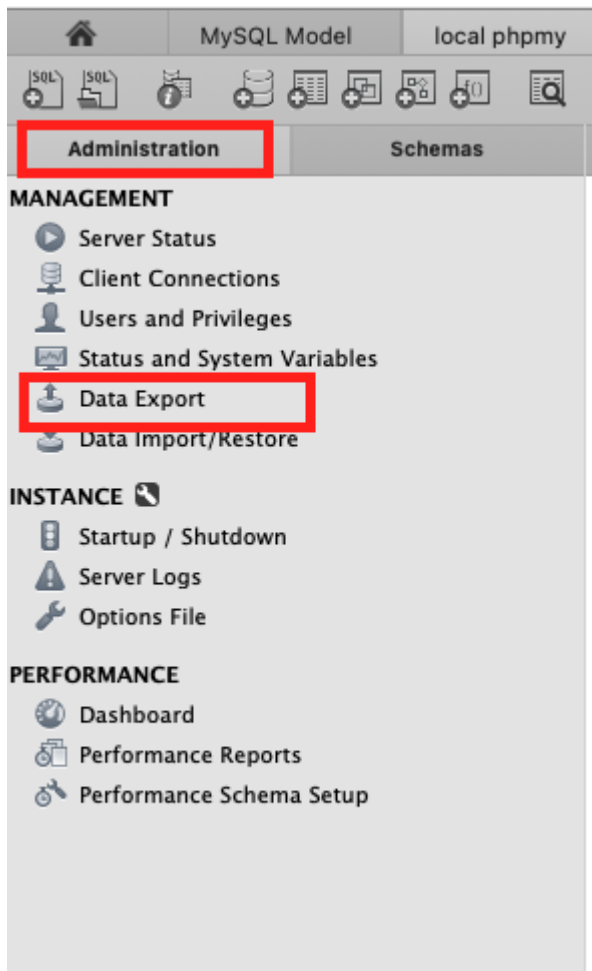
- Extraire les propositions pertinentes de l'énoncé.
- Proposer, sous forme de textes courts, les entités, attributs, associations et cardinalités
- Identifier les dépendances fonctionnelles
- Réaliser un schéma entité-association sur papier (propre) ou à l'aide d'un logiciel de votre choix ( Diagrams.net, ERDPlus, Lucidchart, Dia ou autre) avec types d'entité, associations, cardinalités et annotations
- 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 EA en schéma de table dans le logiciel MySQL WorkBench et sauvegarder votre fichier .mwb pour l'ajouter à votre rendu.
- Générer le code SQL du diagramme et tester la création de la base de données dans MySQL. Sauvegarder dans un fichier **database\_create.sql**.

Vous pouvez réaliser des exports de la structure de votre base, des données, etc. en vous connectant à votre serveur MySQL depuis MySQL Workbench, dans l'onglet *Administration > Data Export* :



- Ajouter des données à la base de données - Cette partie peut se révéler fastidieuse, cependant elle vous permettra de bien comprendre les relations entre vos tables et de vérifier si vous pourrez répondre aux requêtes et contraintes demandées.

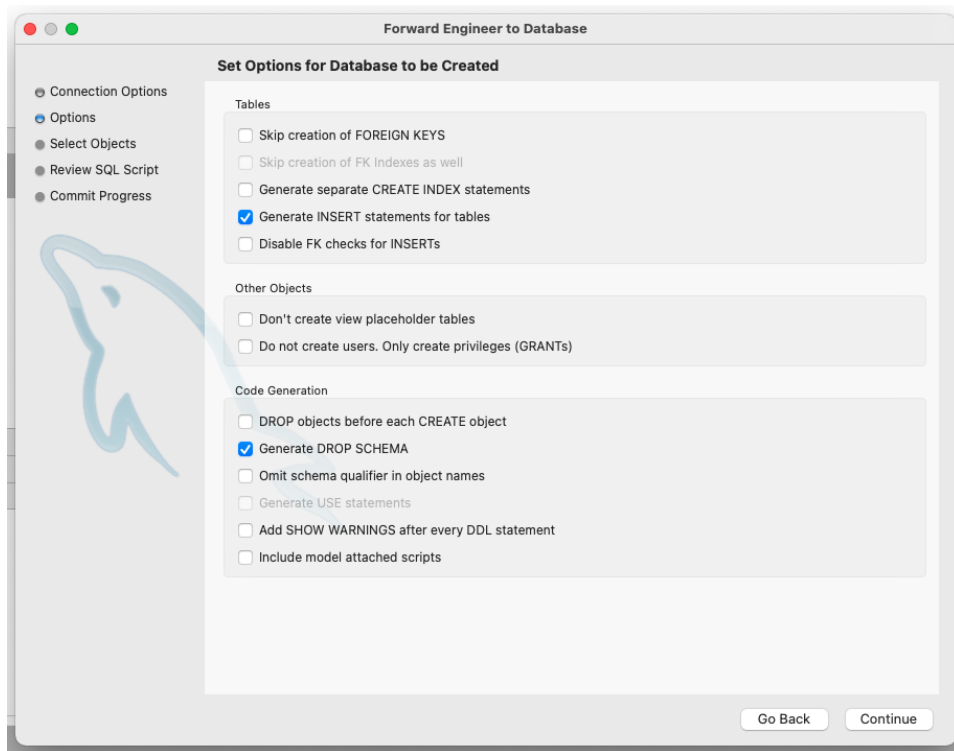
*Vous pouvez ajouter des données manuellement ou bien utiliser l'outil <https://www.mockaroo.com/> pour générer des fausses données. Vous pouvez télécharger les schémas Mockaroo en CSV, les ouvrir dans un fichier texte, puis recopier les données dans l'onglet "insert" de vos tables dans votre modèle MWB.*

*Ces données pourront ensuite être insérées à l'aide du menu Database > Forward Engineer, puis sélectionner les options "Generate INSERT statements for table" et "Drop Schéma". Ceci vous permet de facilement supprimer votre base et la recréer avec vos données. **Attention cependant, toutes vos données seront réinitialisées par ce***

*procédé.*

*N'essayez pas forcément de tout générer avec Mockaroo. Essayez de créer quelques données types qui vous permettront de valider vos requêtes. Par exemple : 5 salariés, 10 candidats, 10 métiers, 20 compétences, 10 offres d'emplois avec divers statuts et liens avec les salariés et candidats, puis vous pouvez terminer à la main pour mettre en relation des candidats avec des offres d'emploi.*

- Créer la base de données dans MySQL à l'aide de MySQL WorkBench (Database > Forward Engineer).



- 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 SQL - 10pts

Merci d'ajouter des commentaires dans votre code SQL avec le numéro et l'énoncé de chaque requête, ainsi que des explications si vous le jugez utile.

1. Une liste des candidats et des offres auxquelles ils ont postulé.

2. Proposez une ou plusieurs requêtes pour afficher toutes les informations d'un candidat (Informations personnelles, diplômes, métiers, compétences, offres d'emploi, etc.)
3. Lister les candidats qui ont postulé à toutes les offres d'emploi
4. Calculer la somme des commissions perçues par l'agence (toutes périodes confondues)
5. Même chose que la 4, mais cette fois ci groupées par mois
6. Même chose que 4 et 5, mais cette fois ci calculer les commissions par salarié et non par l'agence
7. Lister les compétences qui ne sont pas rattachées à des candidats
8. Lister les candidats potentiels (qui n'ont pas encore candidaté) pour une offre d'emploi donnée (c.à.d les candidats qui sont liés aux métiers ou compétences requises par l'offre d'emploi)
9. Ajouter les candidatures pour les candidats listé en requête 8 (avec une ou plusieurs requêtes imbriquées ou en utilisant des vues, il ne faut pas lister manuellement les identifiants des candidats dans la ou les requêtes d'insertion)
10. Fermer une offre d'emploi et l'attribuer à un candidat

*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, procédures (routines) et déclencheurs (triggers) - 10pts

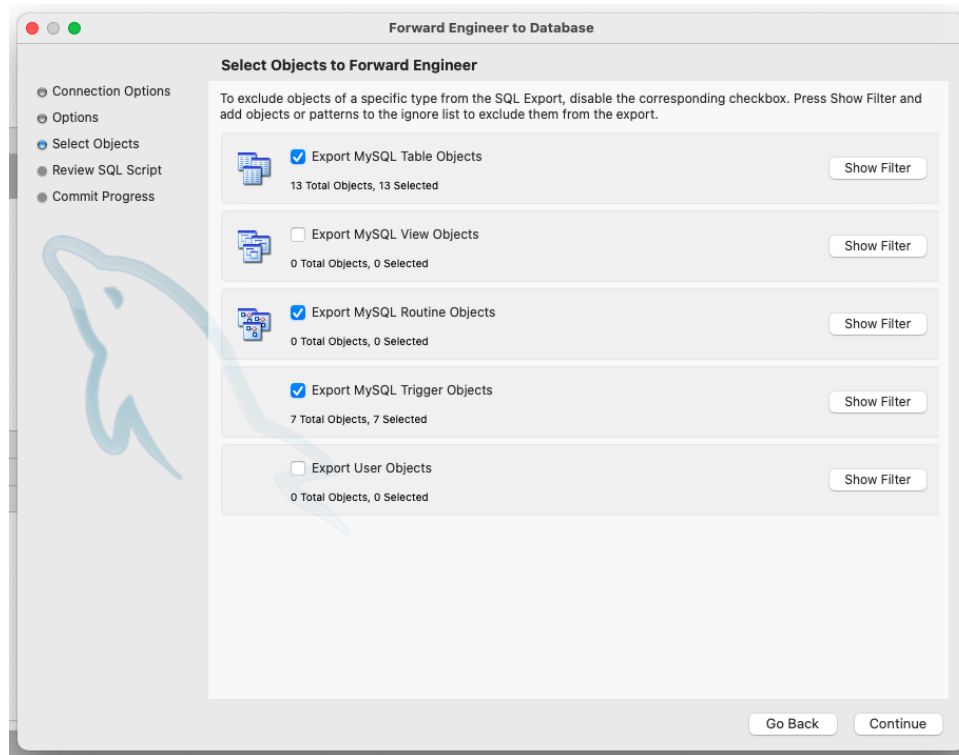
Merci d'ajouter des commentaires dans votre code SQL avec le numéro et l'énoncé de chaque contrainte, ainsi que des explications si vous le jugez utile.

*Conseil : Pour rédiger les triggers et procédures, vous devrez effectuer des recherches sur internet. Effectuez des recherches en anglais du type "mysql trigger reject delete" (par exemple).*

*Les triggers peuvent être ajoutés sur les tables directement dans votre modèle MWB. Cliquez sur une table puis allez dans l'onglet "Triggers". En créant les triggers dans MWB, vous n'avez pas besoin de changer le délimiteur de ; en \$\$.*

Pour les procédures, vous pouvez les ajouter en cliquant sur "Add Routine" en dessous de vos tables.

Une fois créés, vous pouvez synchroniser votre base en utilisant le menu Database > Forward Engineer comme vous l'avez fait pour les insertions dans la partie 2.



En lançant un Forward Engineer avec ces options, vous supprimez votre base, puis la créez à nouveau avec toutes vos insertions, procédures et triggers. Les triggers seront déclenchés pendant les insertions, vous permettant de déjà faire certaines vérifications. **Attention cependant, ce procédé va réinitialiser toutes vos données.**

En utilisant des contraintes statiques ou dynamiques, des procédures (routines) et des déclencheurs (triggers), proposez des solutions pour mettre en place les contraintes suivantes :

1. Une offre d'emploi ne peut pas être supprimée, uniquement archivée (statut "fermée"). Il faut donc créer un trigger pour bloquer la suppression d'une part.
2. D'autre part, lorsqu'une offre d'emploi change de statut (de ouverte à fermée), on doit enregistrer la date du jour comme date de fermeture. Il faut donc créer un trigger avant la mise à jour de la table offres, pour modifier la date de fermeture ssi le statut passe de *ouverte* à *fermée*. (on doit utiliser le trigger before\_update car il n'est pas possible de mettre à jour une ligne avec after\_update. [Une explication d'exemple](#)).



3. Un candidat peut demander la suppression de ses données personnelles. Cependant, un candidat ne devrait pas être supprimé de la base de données, ses données non personnelles doivent être conservées pour archive. La suppression d'un candidat devra se traduire par la suppression de ses données personnelles uniquement. Selon votre conception, vous devrez créer un trigger, mettre des valeurs à null ou les deux.
4. Les candidats ne peuvent pas obtenir une offre d'emploi s'ils ne sont pas disponibles (Les candidats sont considérés non disponibles s'ils ont obtenu une offre d'emploi et qu'elle n'a pas de date de fin ou que la date de fin de l'offre est supérieure à la date actuelle). Selon votre conception, plusieurs solutions sont possibles.
5. Une candidature ne peut être obtenue si elle a déjà été obtenue par un autre candidat.
6. Rédiger une procédure (routine) qui change le statut des candidats à disponible si leur emploi actuel s'est terminé.
7. Les candidats doivent être âgés d'au moins 16 ans. On devra donc vérifier à l'insertion d'un nouveau candidat que sa date de naissance est bien supérieure à 16 ans par rapport à la date du jour.

*Vous devez proposer des requêtes permettant de vérifier que les contraintes fonctionnent, ou bien m'indiquer comment les tester. Ajoutez vos requêtes de tests des contraintes dans un fichier **database\_triggers.sql** et ajoutez le fichier à votre archive de rendu.*

## 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, essayez 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) exécution des requêtes 4) ajout et tests des vues, procédures ou triggers.

Je vais corriger 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.

# 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.