

CIVIL 307 BIM

Table des matières

1.	LIER UNE BASE DE DONNÉES AVEC UN LOGICIEL BIM	2
1.1.	LIEN DIRECT ENTRE REVIT ET UNE BASE DE DONNÉES.....	2
1.2.	MARCHE À SUIVRE	2
2.	INTRODUCTION AUX BASES DE DONNÉES	5
2.1.	INTRODUCTION À LA GESTION D'UNE BASE DE DONNÉES	5
2.2.	LES PREMIERS PAS AVEC UNE BASE DE DONNÉES : EXEMPLE CONCRET N°1.....	5
2.3.	EFFECTUER DES REQUÊTES DANS UNE BASE DE DONNÉE : EXEMPLE CONCRET N°2.....	9

1. LIER UNE BASE DE DONNÉES AVEC UN LOGICIEL BIM

- Support informatique : Maquette Revit 2024 de l'HEPIA (fichier HEPIA Ailes AB.rvt)

Le but de ce chapitre est de créer les nomenclatures des éléments de structure en béton et des portes dans Revit puis d'explorer différentes façons d'exporter les données du projet source vers MySQL afin de les exploiter à l'aide de requêtes SQL (Structured Query Language) ou à travers un module externe (add-in).

Nous utilisons un modèle réalisé sur Revit 2024.

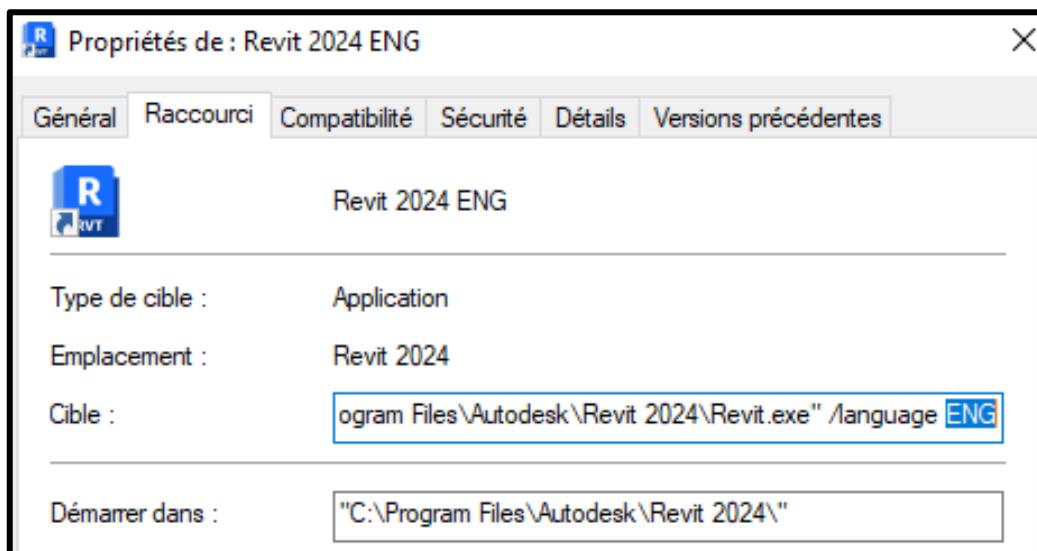
1.1. LIEN DIRECT ENTRE REVIT ET UNE BASE DE DONNÉES

Nous allons exporter les informations de la maquette vers une base de données MySQL et mettre en place des requêtes pour récupérer certaines données. Il existe plusieurs manières de procéder, en passant par le driver Advance ODBC Driver d'Autodesk, les 2 principales étant :

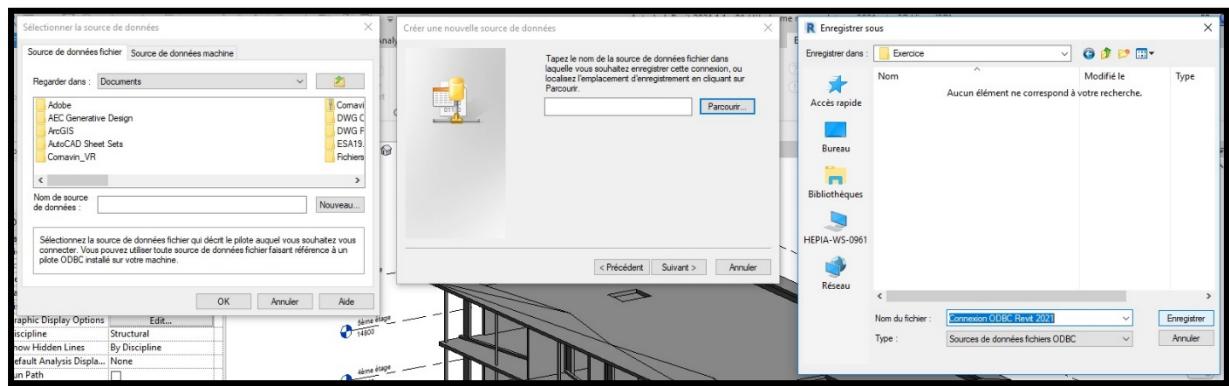
- Connexion au serveur SQL, utilisé principalement si le serveur est hébergé ailleurs, sur un cloud par exemple ;
- **Connexion au serveur MySQL ODBC 8.0 ANSI Driver, celle que nous allons utiliser car nous travaillons en local ;**

1.2. MARCHE À SUIVRE

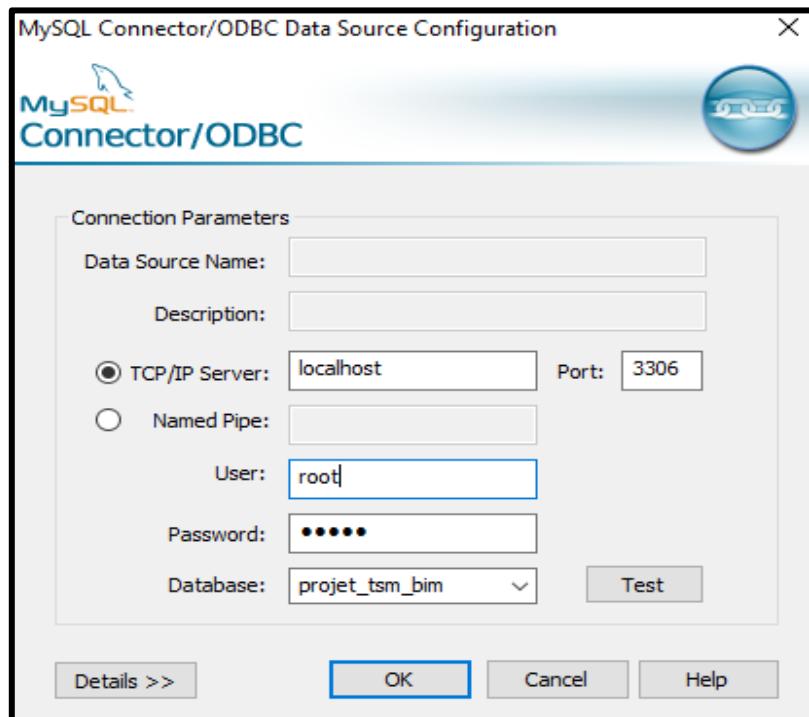
- Vérifier que Revit est bien configuré en anglais, si ce n'est pas le cas vous pouvez changer la langue en faisant un clic droit sur l'icône Revit 2024 sur votre bureau > Propriétés, et taper ENG dans la Cible puis Appliquer :



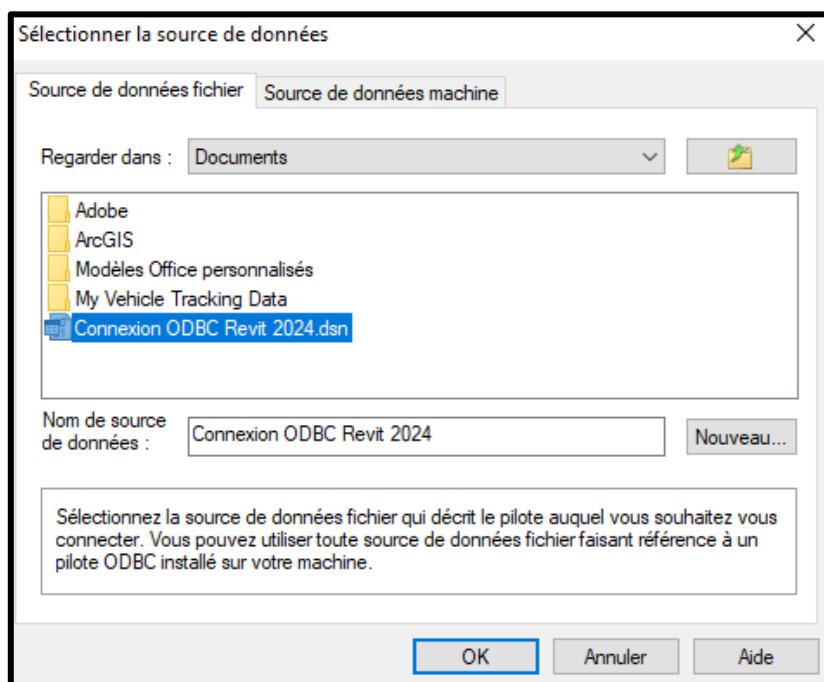
- Ouvrir MySQL Workbench 8.0 CE, cliquer sur « Local instance MySQL80 » et taper **create database projet_tsm_bim** dans la fenêtre centrale. Cliquer sur l'icône  pour exécuter la commande puis sur l'icône Refresh à droite de **SCHEMAS** ;
- Ouvrir Revit, charger la maquette HEPIA Ailes AB.rvt, cliquer sur File > Export > ODBC Database ;
- Cliquer sur Nouveau à droite du champ « Nom de source de données » puis sélectionner MySQL ODBC 8.0 ANSI Driver et cliquer sur suivant ;
- Cliquer sur Parcourir à droite du champ vide, choisir l'emplacement où vous voulez enregistrer le fichier DSN, nommez-le *Connexion ODBC Revit 2021* par exemple et cliquer sur Enregistrer puis sur Suivant et Terminer ;



- Remplir les champs de la fenêtre MySQL Connector/ODBC selon la figure suivante avec le password « admin » :



- Cliquer sur Test puis sur OK si la connexion est opérationnelle.
- Si l'export ne se lance pas il faut sélectionner le fichier DSN dans la fenêtre, cliquer sur OK, retaper le password dans la fenêtre suivante et cliquer sur OK ;



- Si l'export est bien lancé l'anneau de chargement bleu apparaît dans Revit ;

2. INTRODUCTION AUX BASES DE DONNÉES

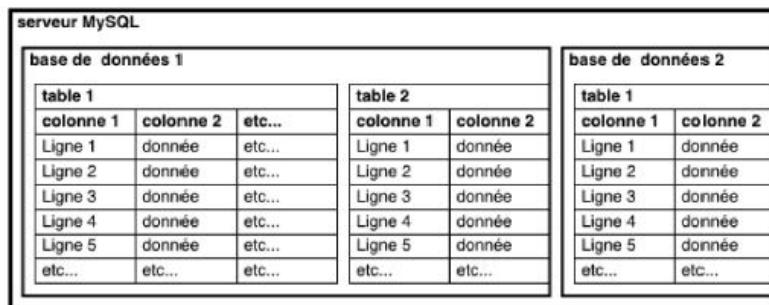
- Durée : 150'.
- Support informatique : MySQL
- Support documentaire : Liste des query dans le fichier .txt en annexe (01-Commandes pour remplir les champs - Exercice db_portique & 02-Autres requêtes).

2.1. INTRODUCTION À LA GESTION D'UNE BASE DE DONNÉES

- Durée : 30'.

SQL est un langage informatique permettant d'interroger des bases de données par le biais de requêtes sur le principe du modèle relationnel. Nous allons utiliser le logiciel MySQL Workbench pour s'initier à la gestion d'une base de données à travers un exemple simple et concret.

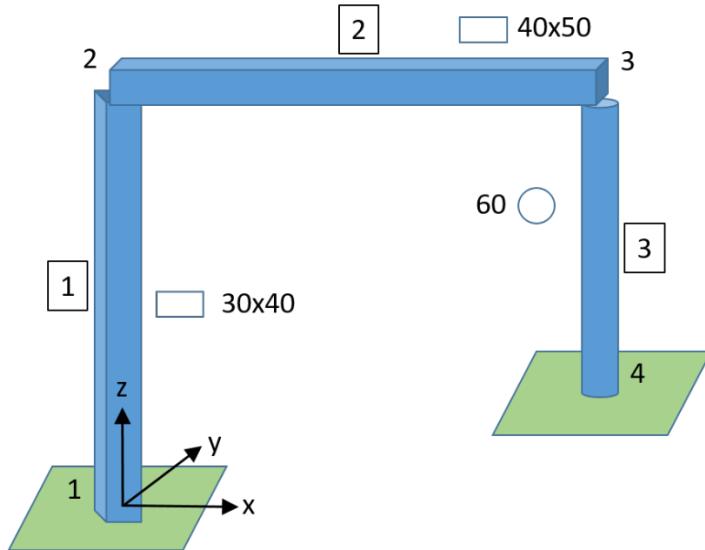
La figure ci-dessous présente la structure d'une base de données en table :



2.2. LES PREMIERS PAS AVEC UNE BASE DE DONNÉES : EXEMPLE CONCRET N°1

- Durée : 120'.

Nous souhaitons mettre sous forme de base de données les informations liées à la description du portique présenté dans le cours :



Les informations qui nous intéressent dans cet exemple sont :

- Le positionnement de chaque élément dans l'espace (points de départ et d'arrivée) ;
- Les caractéristiques des éléments (géométrie, type de matériau) ;

Nous devons ensuite préparer la structure des tables (nom des champs, type de données stockées dans chaque champ, etc.) qui vont rassembler ces informations et définir les relations entre les différentes tables.

Réfléchissez aux différents champs qui devront figurer dans chacune des tables suivantes en appliquant la logique de l'exemple présenté pendant le cours et en l'adaptant à notre portique :

- Éléments (frame) ;
- Nœuds (nodes) ;
- Sections (section) ;
 - rectangle
 - circular
- Matériaux (material) ;
 - concrete
 - wood

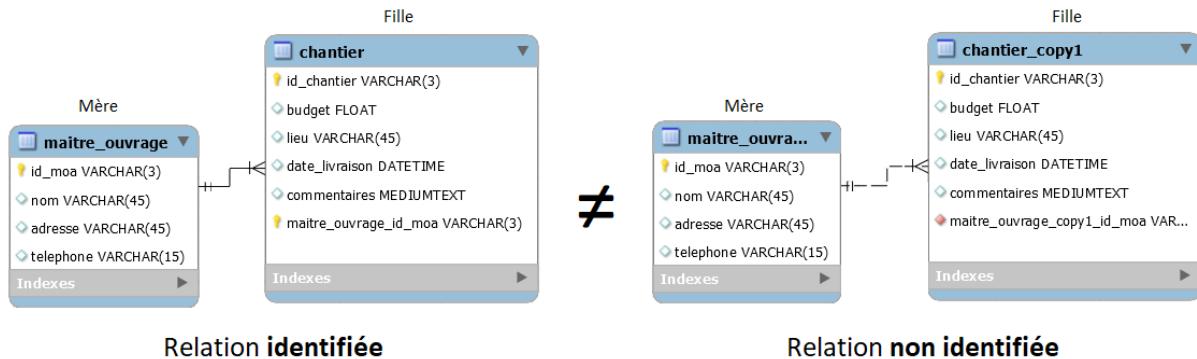
Une fois que la structure de chaque table a été définie, il faut réfléchir à la nature des relations entre les différentes tables. La liste suivante présente les 3 types de relation possibles entre deux tables en fonction de la cardinalité de la relation :

- **Relation de type 1:1** → Chaque objet de la première table est associé à un seul objet dans la deuxième table et chaque objet de la deuxième table est associée avec un objet de la première table uniquement.
 - Exemple (Une personne peut garer son véhicule sur un seul emplacement dans un Parking)

- **Relation de type 1:n** → Chaque objet de la première table peut être associé à un ou plusieurs objets dans la deuxième table, mais chaque objet de la deuxième table n'est associé qu'à un seul objet de la première table.
 - Exemple (Un enseignant peut dispenser plusieurs cours, mais chaque cours est dispensé par uniquement un enseignant)
- **Relation de type n:m** → Chaque objet de la première table possède une relation avec plusieurs objets de la deuxième table et chaque objet de la deuxième table est relié avec plusieurs objets de la première table
 - Exemple (Les étudiants sont inscrits à plusieurs cours et les cours sont suivis par plusieurs étudiants)

Une relation peut être soit **identifiée** (caractérisée par une ligne continue), soit **non-identifiée** (ligne discontinue) :

- Si la relation est identifiée, la clé primaire de la table mère sera clé primaire dans la table fille
- Si la relation est non-identifiée, la clé primaire de la table mère sera clé **étrangère** dans la table fille



En vous référant à l'exemple du portique, définissez la cardinalité des relations entre les tables suivantes :

- La table frame avec la table material ;
- La table frame avec la table section ;

Lorsque la relation entre deux tables est du type **1:1**, il est possible de rassembler les informations réparties entre les deux tables dans une nouvelle table ou vue à l'aide d'un champ qui contient des valeurs identiques dans chacune des tables. **Le processus décrit est une jointure (INNER JOIN).** Il peut être appliqué entre les tables material et concrete, ou entre les tables section et rectangle.

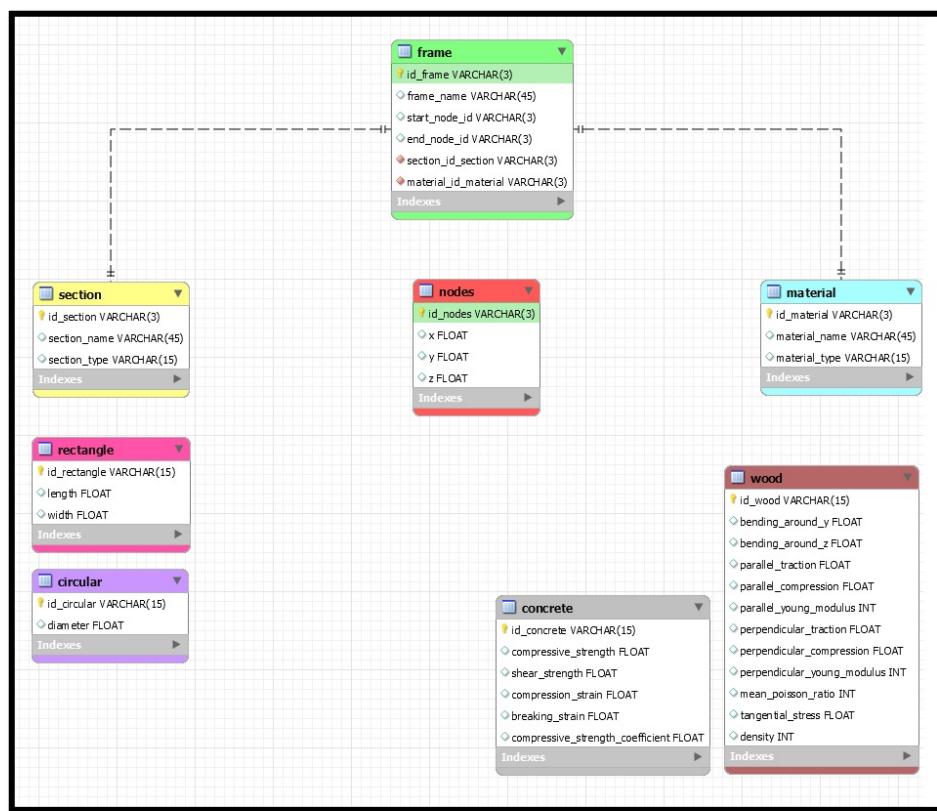
Lorsque la relation entre deux table est différente, il faut soit intégrer les clés primaires des objets de l'une des tables comme clés étrangères dans l'autre table si la relation est du type **1:n**, soit créer une

nouvelle table qui relie chaque clé primaire de la première table avec les clés primaires de la deuxième table si la relation est du type **n:m**. Ce processus s'appelle la **mise en relation**.

Dans le cas d'une relation du type **1:n**, la jointure peut être effectuée de manière directe dans un sens, depuis la table qui contient plusieurs entités liées à une entité unique dans l'autre table. Si la relation est du type **n:m**, la jointure directe est impossible. Il faut alors passer par une table intermédiaire appelée **table de jointure**.

Créer une nouvelle base de données *db_portique*, puis créer les tables.

On peut ensuite compiler (Database > Reverse Engineer), puis explorer (Database > Forward Engineer) les tables ainsi créées.



2.3. EFFECTUER DES REQUÊTES DANS UNE BASE DE DONNÉE : EXEMPLE CONCRET N°2

- Durée : 30'.
- Support informatique : MySQL.

On peut retourner sur MySQL Workbench, faire un clic droit dans schemas puis refresh all, puis dérouler la base de données pour voir que l'export a bien marché.

Les données peuvent alors être examinées à travers des requêtes dans le logiciel de gestion de la base de données (type MySQL).

```
/*** Afficher la table des poteaux structurels ***/  
select * from `structural column types`  
  
/*** Sélectionner les colonnes de 3.07m de haut ***/  
select id from `structural columns` where `Length`='3.07'  
  
/*** Sélectionner les colonnes appartenant à la famille Concrete Round ***/  
select id, `Family Name` from `structural column types` where `Family Name` ='Concrete Round'  
  
/*** Faire la liste des colonnes au Rez Supérieur ***/  
select levels.name, `structural columns`.id, `structural columns`.`type id`, `structural column types`.'Family Name'  
from levels, `structural columns`, `structural column types`  
where levels.name='00 - Rez Supérieur' and `structural columns`.`base level` = levels.id and  
`structural columns`.`type id`=`structural column types`.id  
/*** Calculer le prix de 5 colonnes 424314 ***/  
update `structural column types` set cost=10 where id=424314  
select cost*5 from `structural column types` where id=424314
```