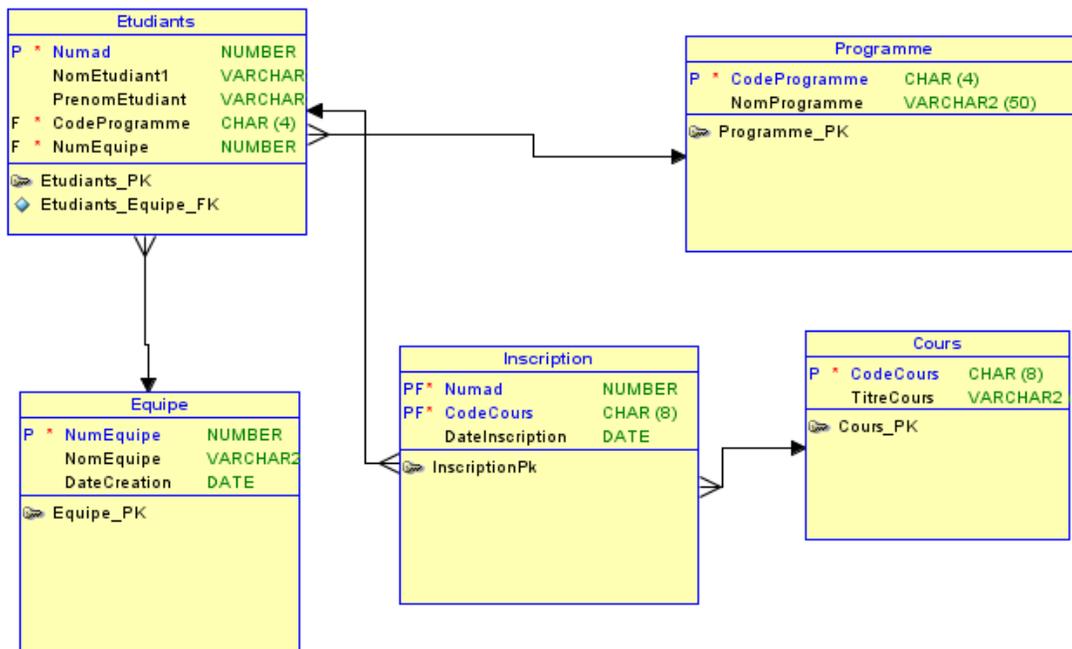


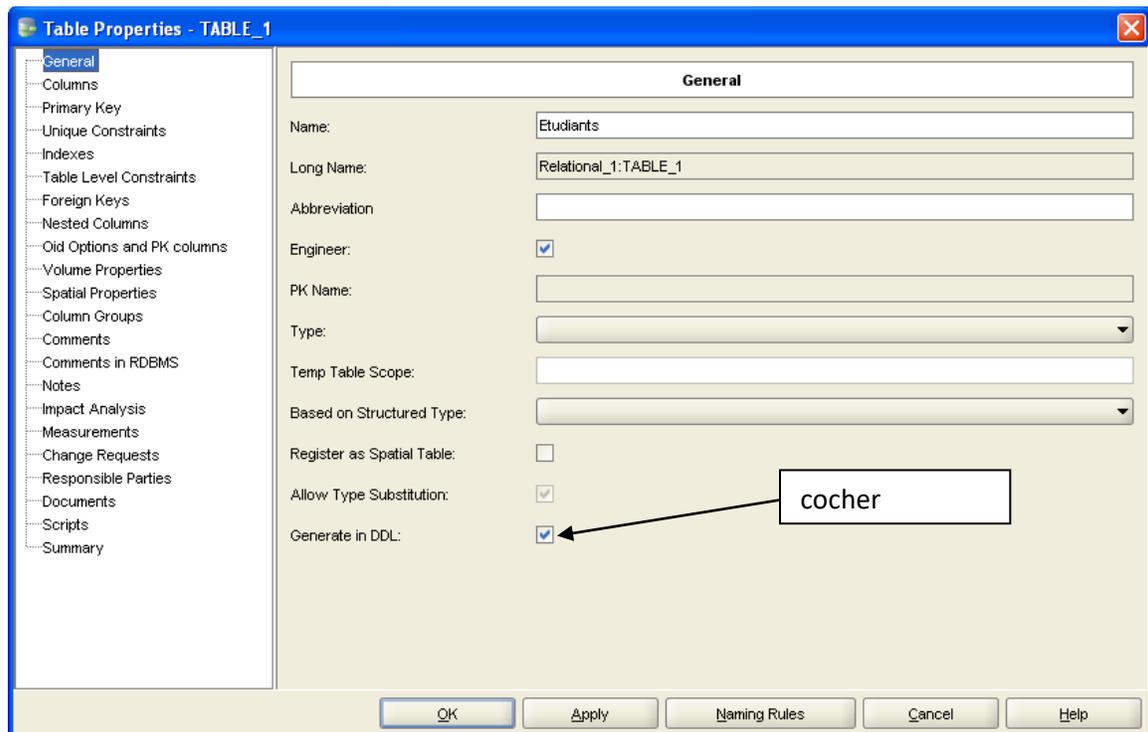
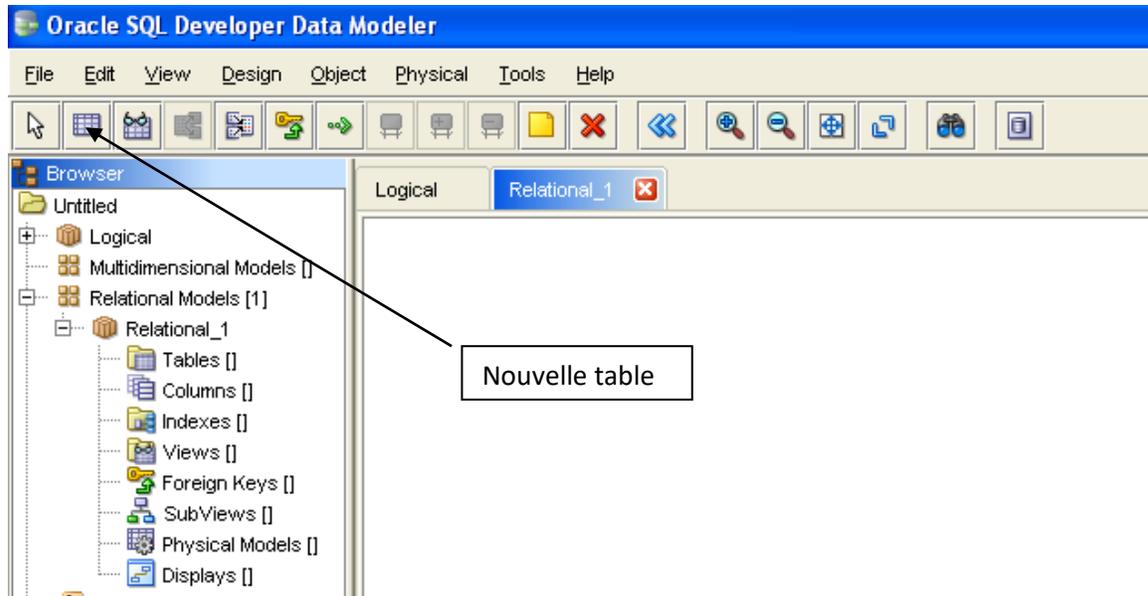
Utilisation du modèle relationnel sous SQL Developer Data Modeler et la génération du code SQL

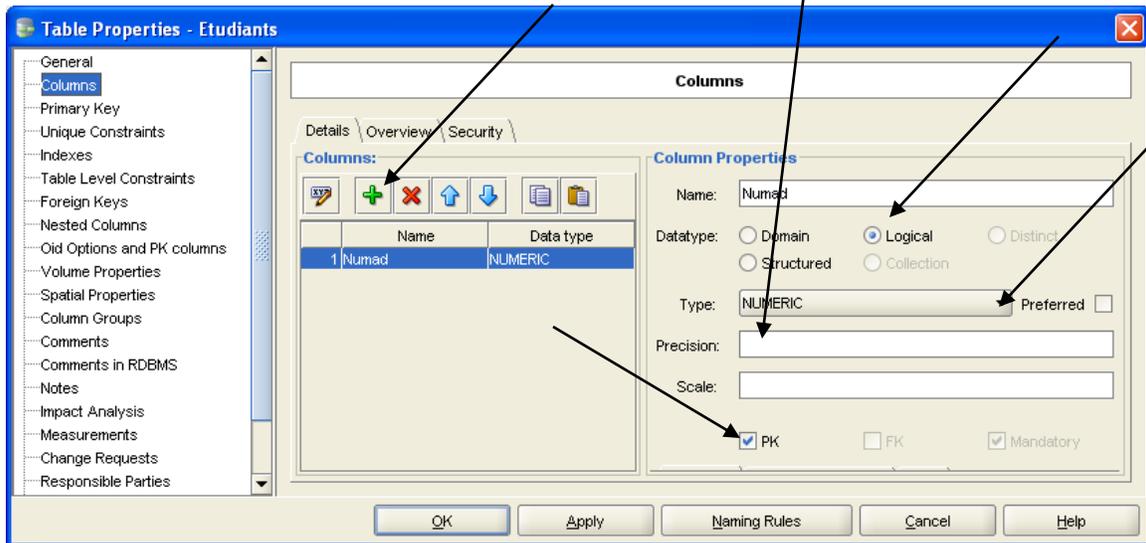
Nous voulons construire le modèle relationnel suivant :

1. La table Programme contient une clé primaire sur l'attribut CodeProgramme.
2. La table Equipe contient une clé primaire sur l'attribut NumEquipe
3. La table Cours contient une clé primaire sur l'attribut CodeCours.
4. La table étudiant contient :
 - a. Une clé primaire sur l'attribut Numad
 - b. Une clé étrangère sur l'attribut CodeProgramme et faisant référence à la clé primaire de la table Programme.
 - c. Une clé étrangère sur l'attribut NumEquipe faisant référence sur la clé primaire de la table Equipe.
5. La table inscription contient :
 - a. Une clé composée de (Numad,CodeCours)
 - b. Une clé étrangère sur l'attribut Numad faisant référence à la clé primaire de la table Etudiants.
 - c. Une clé étrangère sur l'attribut CodeCours faisant référence à la clé primaire de la table Cours.



Démarrer SQL Data Modeler sur le bouton droit de Relational Models, choisir New Relational Model





Pour mettre une clé étrangère vous avez le choix de procéder comme suit :

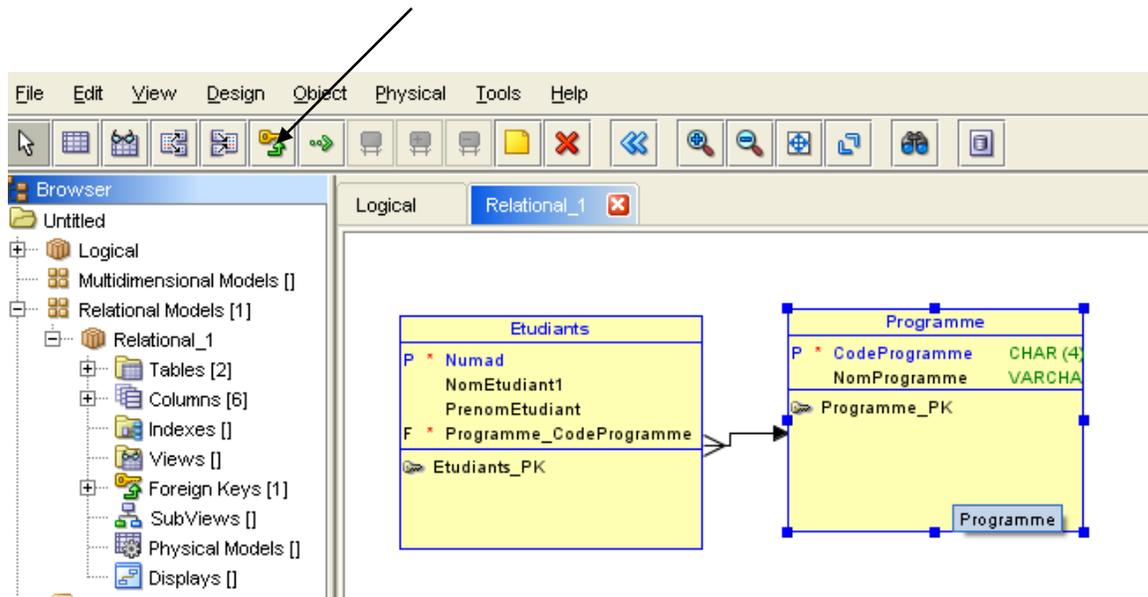
Cas 1 : l'attribut qui correspond à la clé étrangère n'est pas dans la table concernée : Le code programme n'est pas dans la table Etudiants

Etudiants	
P *	Numad NUMBER
	NomEtudiant1 VARCHAR
	PrenomEtudiant VARCHAR
Etudiants_PK	

Programme	
P *	CodeProgramme CHAR (4)
	NomProgramme VARCHAR
Programme_PK	

Utiliser le symbole correspondant à la clé primaire (voir figure suivante). Vous glissez la clé primaire de la table programme dans la table étudiants. Vous aurez le schéma suivant.

En tout temps, vous pouvez modifier les propriétés de la relation en cliquant dessus avec le bouton droit et choisir Propriétés.



En utilisant les propriétés de la table ETUDIANTS , vous pouvez renommer Programme_codeProgramme par CodeProgramme (par le même nom que la clé primaire référencée).

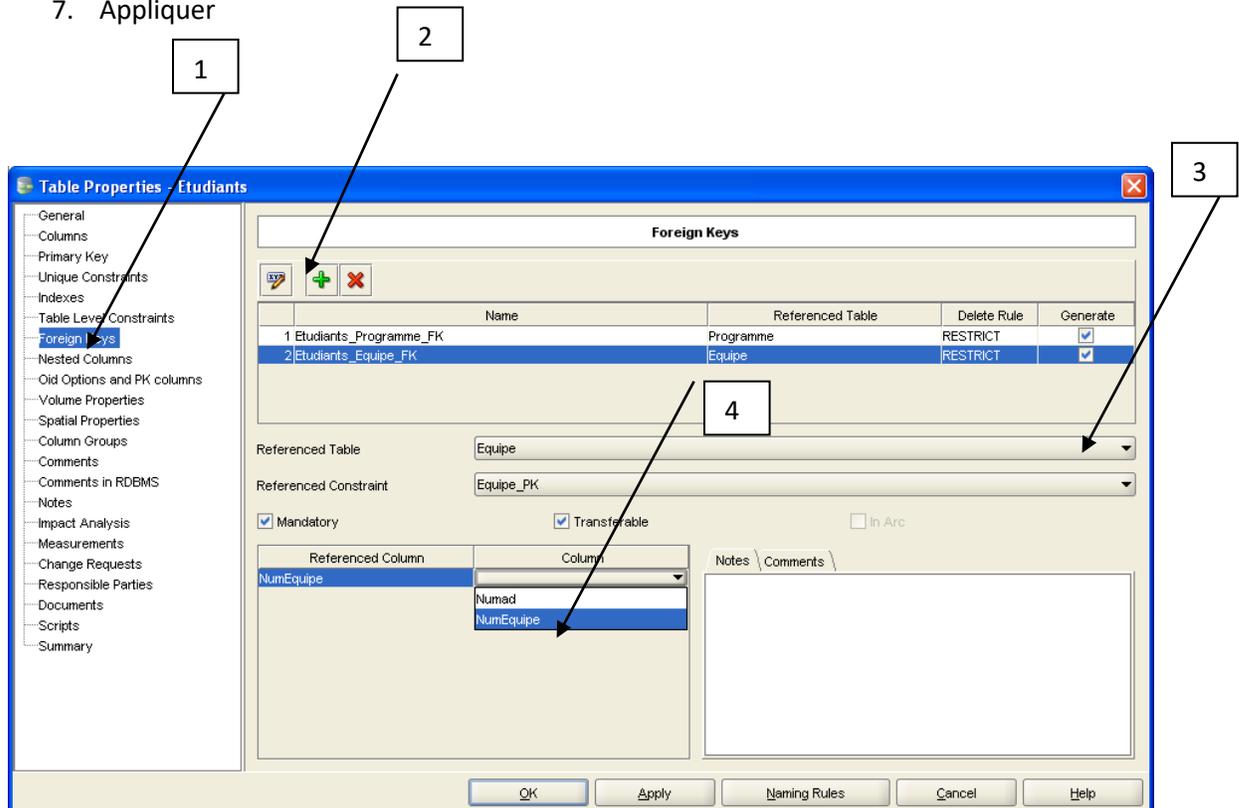
The screenshot shows the 'ForeignKey Properties - Etudiants_Programme_FK' dialog box. The 'General' tab is active. The fields are as follows:

Property	Value
Name:	Etudiants_Programme_FK
Table:	Etudiants.Etudiants_Programme_FK
PK / UK Index:	Programme.Programme_PK
Delete Rule:	RESTRICT
Source Table Synonym:	Programme
Target Table Synonym:	Etudiants
Mandatory:	<input checked="" type="checkbox"/>
Transferable (Updatable):	<input checked="" type="checkbox"/>
Generate in DDL:	<input checked="" type="checkbox"/>
In Arc:	<input type="checkbox"/>

Buttons at the bottom: OK, Apply, Cancel, Help.

Cas 2 : l'attribut qui correspond à la clé étrangère est dans la table concernée : Le NumEquipe est dans la table Etudiants

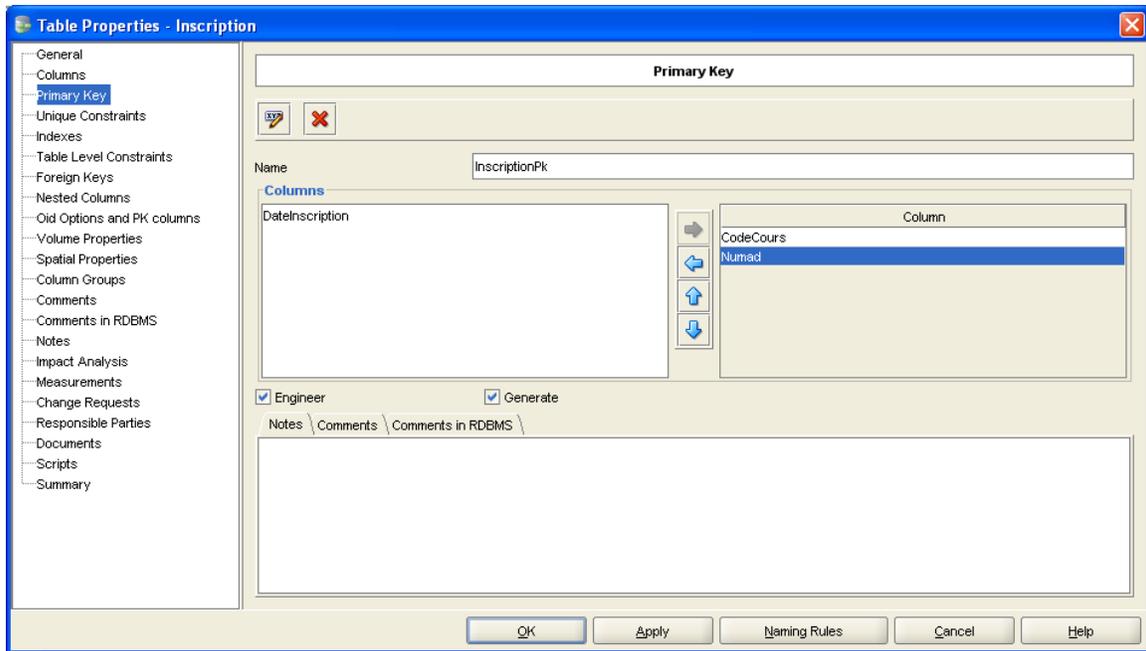
1. Ajouter l'attribut qui va jouer le rôle de la clé étrangère dans la table correspondante :
2. Propriété de la table
3. Choisir foreign Key
4. Ajouter
5. Choisir la table référencée (ici : Équipe)
6. Choisir la colonne dans la table ETUDIANTS
7. Appliquer



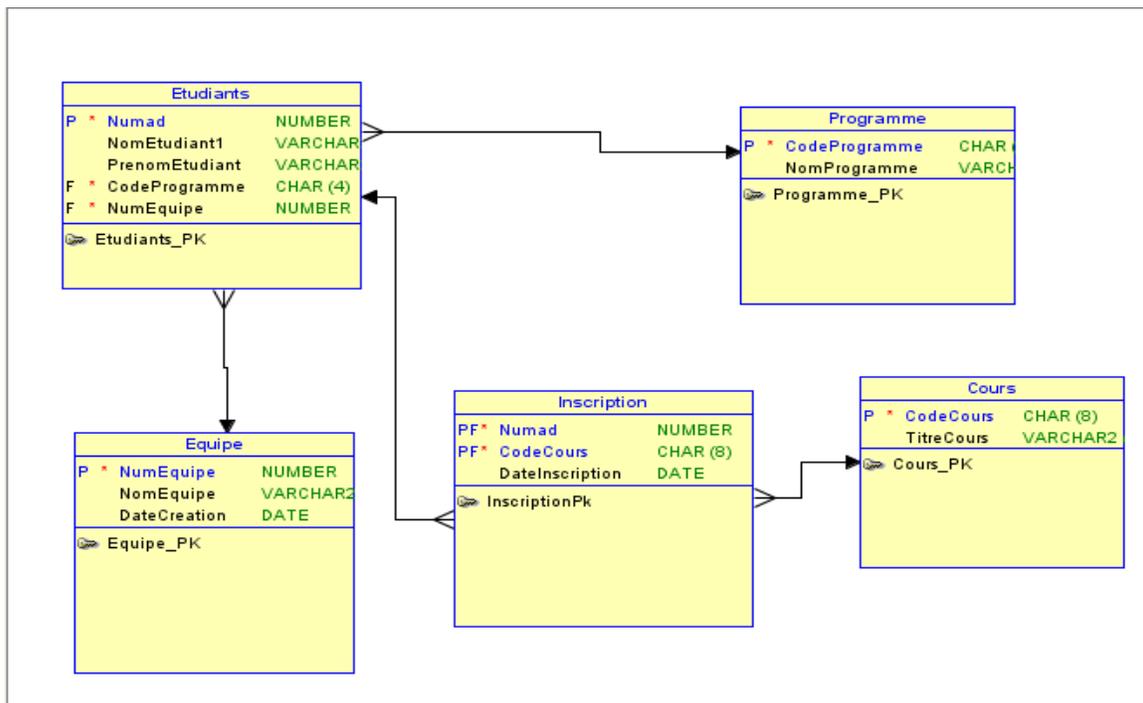
Ajout d'une Clé composée

Ajouter une clé composée revient en deux étapes principales qui sont :

- Ajouter une clé primaire sur les deux attributs (en même temps)
- Ajouter une clé étrangère sur chacun des attributs de la clé composée séparément. Cette étape est importante pour garantir l'intégrité référentielle. L'ordre d'exécution ces deux étapes peut être inversé

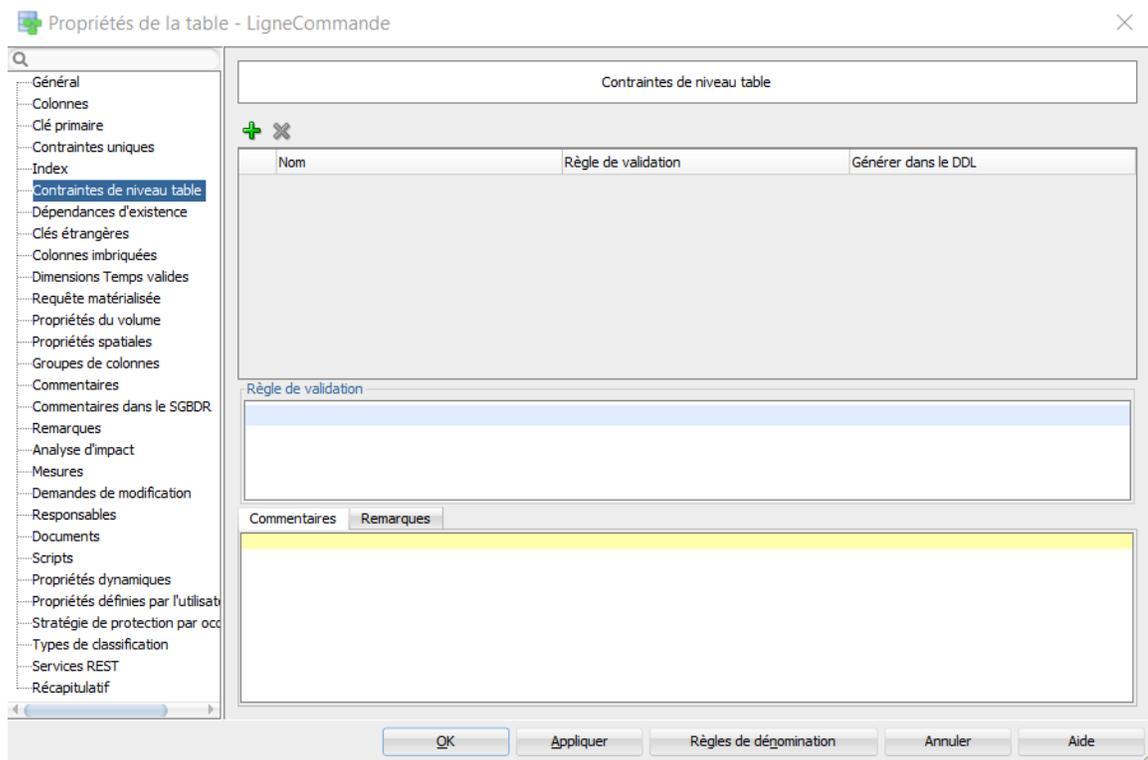


1. Cliquez sur Primary key et ajouter les colonnes que vous voulez qu'elles soient clé primaire.
2. Sélectionnez les champs que vous voulez qu'il soit clé primaire composée
3. Donnez un nom significatif à votre contrainte
4. Cliquez sur OK.
5. Afin de garantir l'intégrité référentielle, ajoutez-les contraintes de clés étrangères pour les attributs de clé primaire. (ici voir le cas 2)

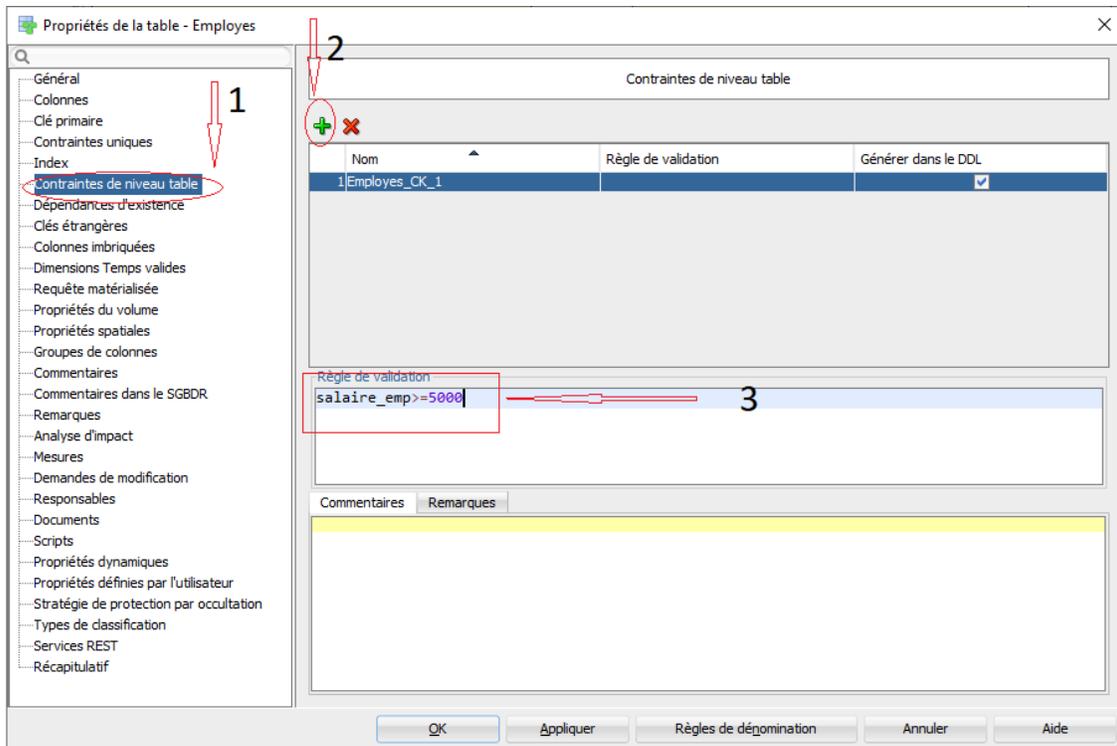


Ajout d'une contrainte CHECK

1. Accéder aux propriétés de la table (bouton droit de la souris, puis Propriétés).
2. Vous obtiendrez la figure suivante :



3. Choisir « Contraintes de niveau de table », puis ajouter (signe + en vert)
4. Vous obtiendrez la figure suivante :



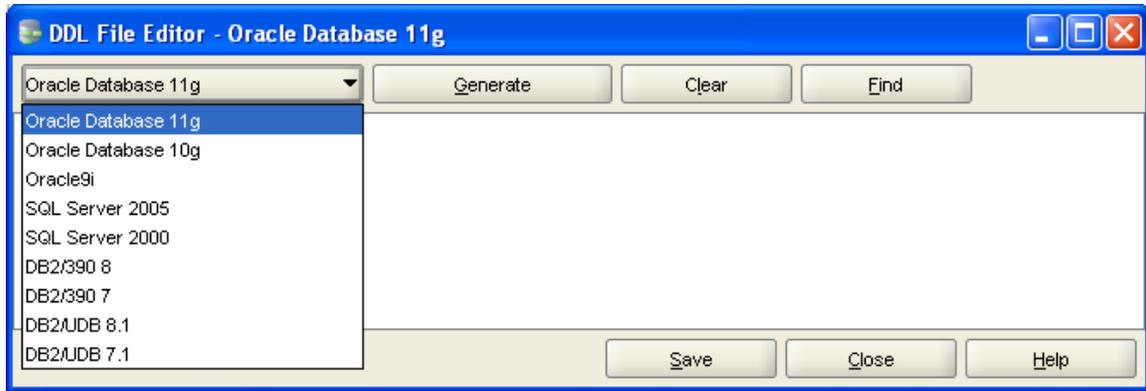
5. À l'endroit « Nom », donner un nom de contrainte significatif.
6. À l'endroit « Règle de validation », écrire votre règle qui correspond à votre contrainte CHECK.
7. Voir exemple plus haut
8. Puis cliquer sur « Appliquer », puis « OK ».
9. **Attention !** Le nom de la variable doit être exactement le même que votre colonne. Il n'y a pas de validation à ce niveau. Si vous avez une erreur, le code SQL sera généré, mais ne pourra pas s'exécuter...

Génération du code SQL

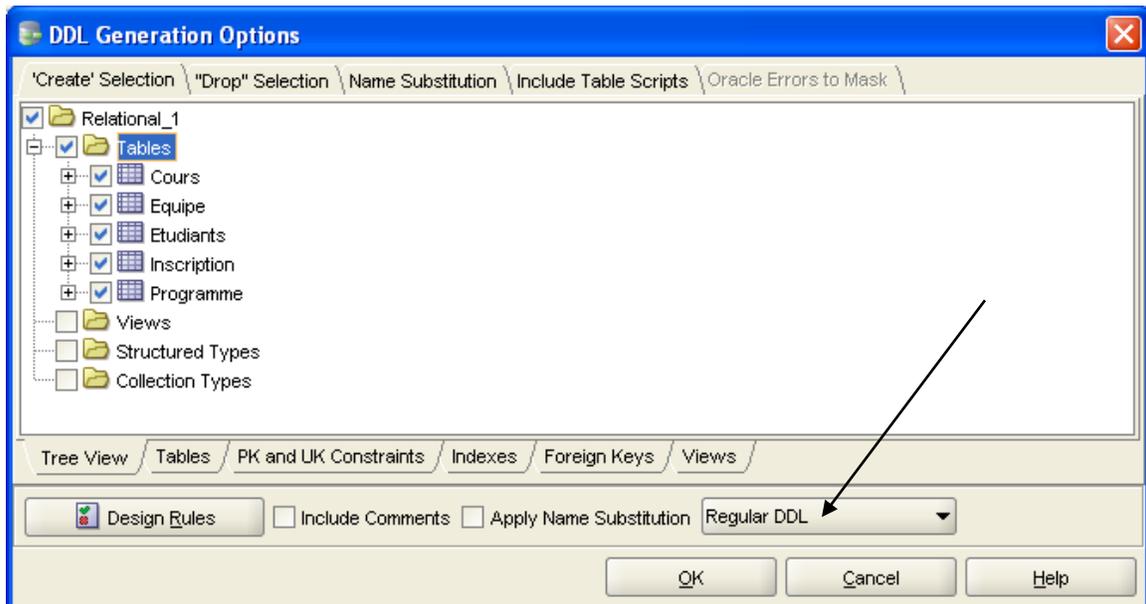
Utilisez le bouton : Generate DDL ou par le Menu View, DDL File Editor.



Vous allez avoir la boîte suivante :

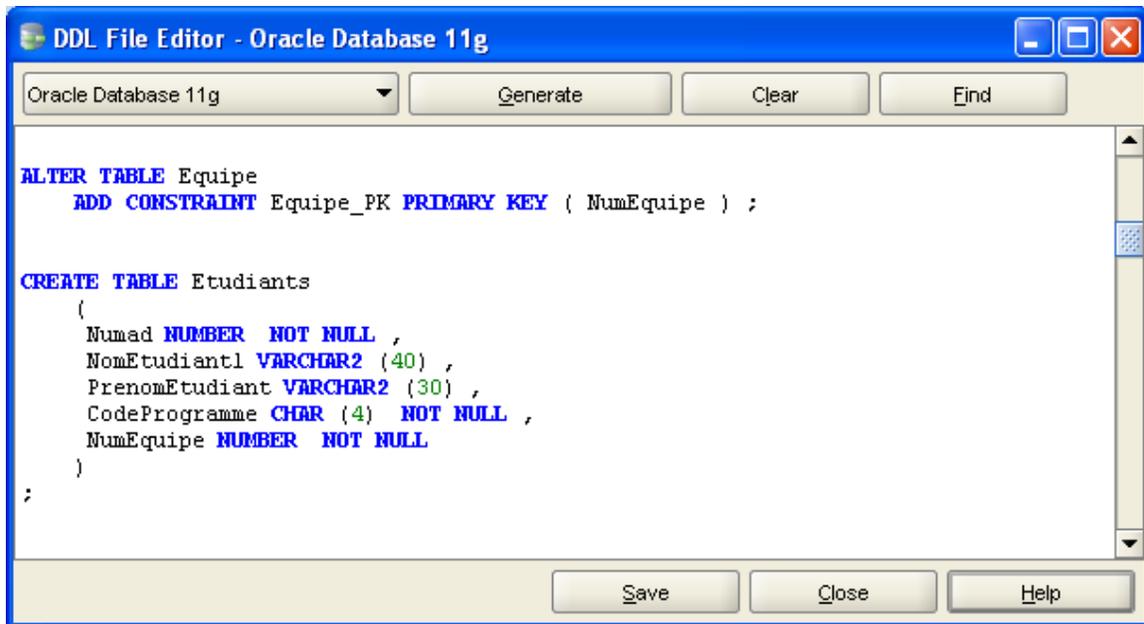


1. Choisir Oracle DataBase 11g, ou Oracle 12c ou Oracle 18 c si disponible
2. Cliquez sur Generate :
Après choisir
3. Choisir Regular DDL
4. **Ne cochez que l'onglet TABLES (voir figure)**
5. Puis sur OK et enfin Save



Il faudra vérifier que tous est OK (selon votre conception)

Vous n'avez qu'à copier votre script et l'exécuter dans SQL Developer.

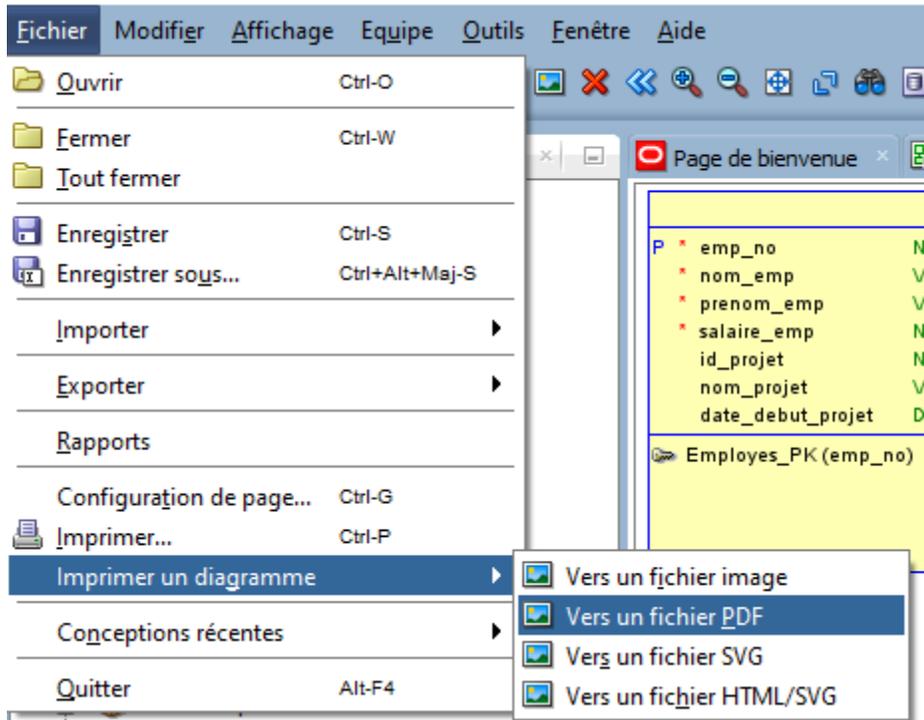


Imprimer son diagramme en PDF ou en PNG

Parfois, vous aurais besoin d'imprimer votre diagramme, pour le lire, le vérifier et le remettre à votre enseignant.

La remise de documents doit toujours se faire dans un format lisible. Pour les modèles de données on recommande un fichier image ou PDF. Pour imprimer son diagramme dans un fichier PDF ou image, rien de plus facile

1. Par le menu Fichier, choisir Imprimer un digramme.
2. Choisir ensuite le format souhaité : Vers un fichier image ou Vers un fichier PDF.
3. Donnez un nom significatif, puis enregistrer.



C'est le fichier image ou PDF obtenu que vous allez remettre aux enseignants lorsque c'est demandé.