



MODÉLISATION ENTITÉ ASSOCIATION

SOMMAIRE

- ☐ CONCEPT DE BASE
- ☐ ENTITÉS
- ☐ ASSOCIATIONS
- ☐ PROPRIETES
- ☐ IDENTIFIANTS
- ☐ CARDINALITES
- ☐ DEMARCHE DE CONCEPTION
- ☐ MODÈLE RELATIONNEL
- ☐ PASSAGE DU MODÈLE ENTITÉ/ASSOCIATION AU
MODÈLE RELATIONNEL



CONCEPT DE BASE

CONCEPT DE BASE

Le modèle E/A est un modèle graphique permettant de décrire et de modéliser les données.

Cette modélisation graphique permet de traduire dans d'autres modèles.

Origine : Travaux de Chen (USA), Tardieu (France), en 1974/75

CONCEPT DE BASE

Le modèle E/A doit son succès à l'utilisation de langage graphique et concepts simples :

- Choses (objets)->entités
- liens entre les choses (objets)-> association
- regroupement des choses de même nature : classes d'entités, classes d'association.



NOTION D'ENTITES

ENTITÉS

Définition : Un objet qui existe dans le monde réel, possède une identité et des propriétés identifiables sans ambiguïté.

Exemple : Un acteur, un étudiant, un employé, etc.

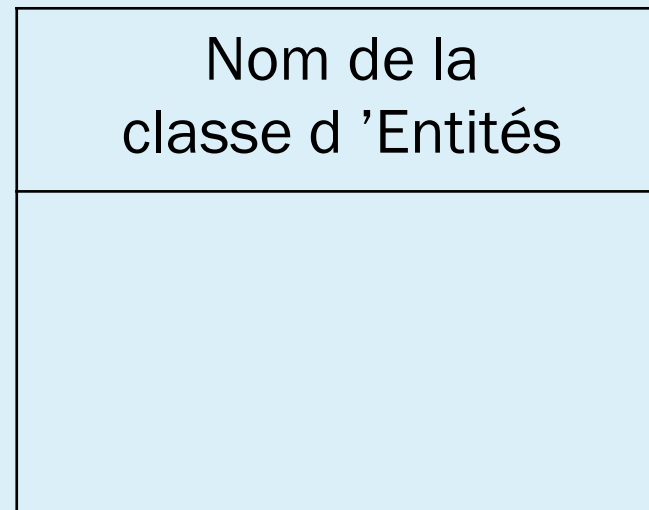
ENTITÉS

Notion de classe d'entités : c'est un regroupement ou un ensemble d'entités de même nature.

Exemple : acteurs, étudiants, employés, etc.

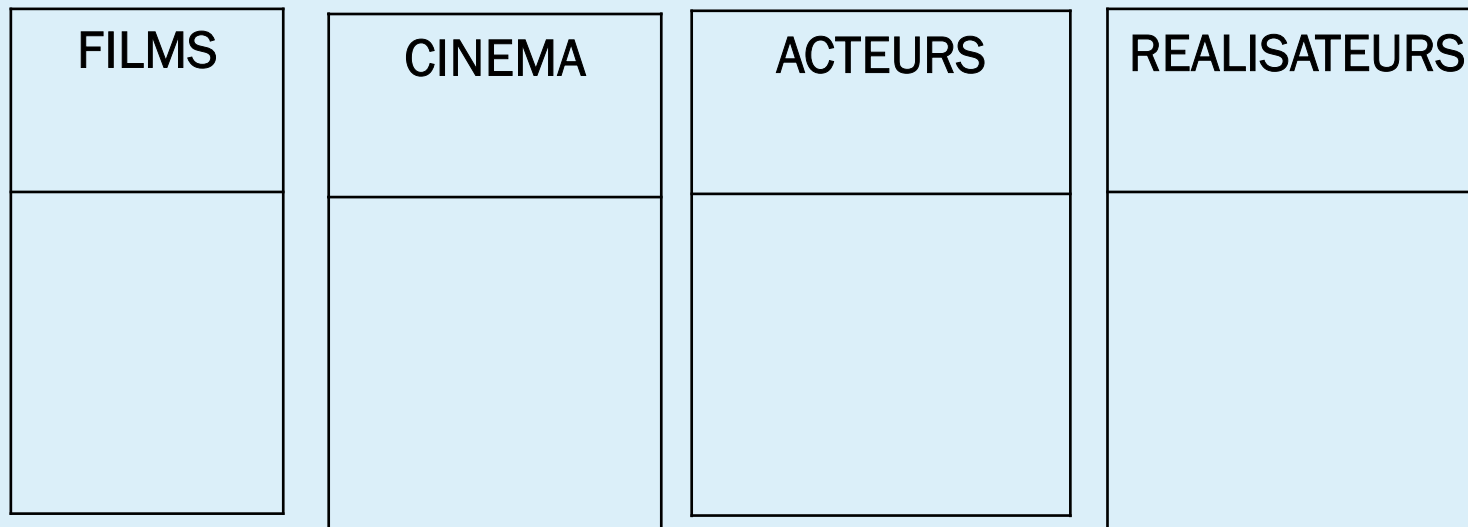
ENTITÉS

Formalisme de classe d'entité :
représentation graphique



ENTITÉS

Exemple de 4 classes d'entités :



Par abus de langage, Entité = Classe d'entités.



NOTION D'ASSOCIATIONS

ASSOCIATIONS

Définition : C'est un lien entre 2 ou plusieurs entités.

Exemple :

- Le célèbre Gohou a joué dans ma famille.
- Demba parle le Wolof

ASSOCIATIONS

Classe d'associations : lien entre 2 ou plusieurs classes d'entités.

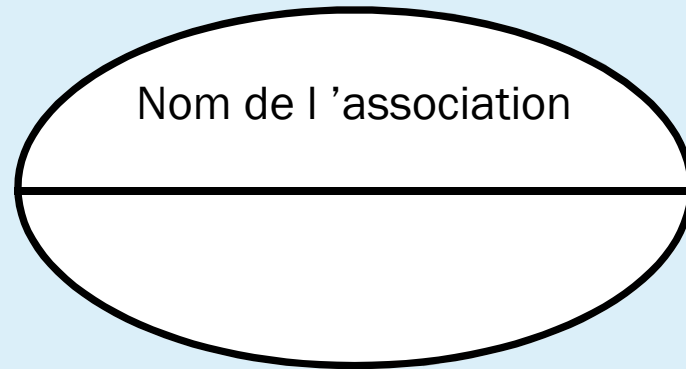
Exemples :

- Jouer (entre FILMS et ACTEURS)
- Affiche (entre CINEMA et FILMS)
- REALISE (entre REALISATEURS et FILMS)

Par abus de langage, association = Classe associations.

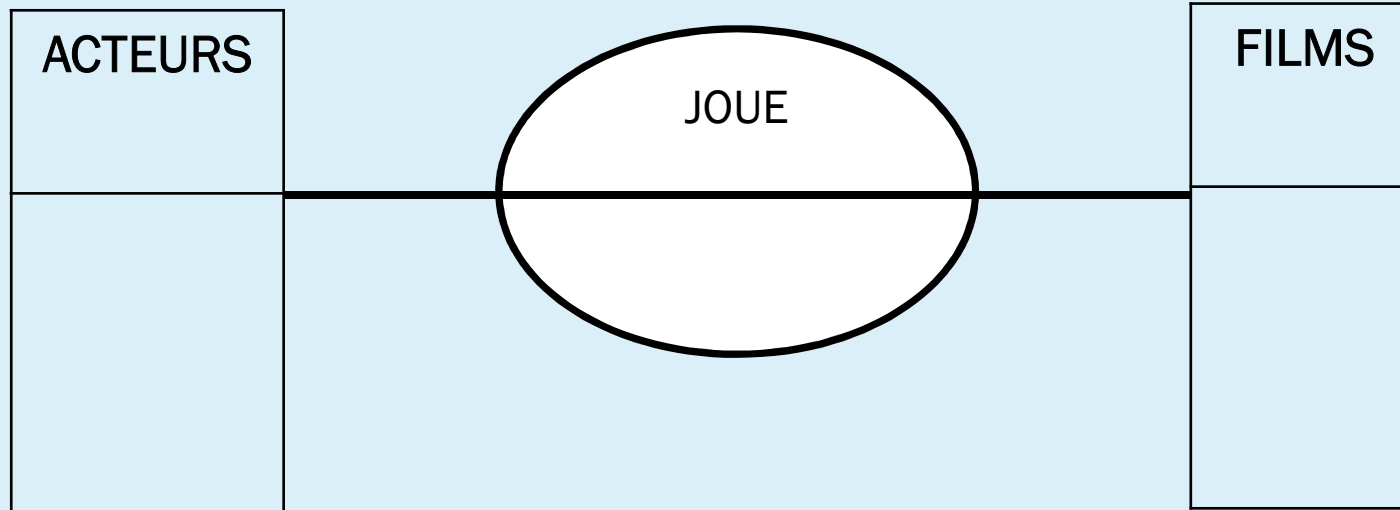
ASSOCIATIONS

Formalisme graphique :



ASSOCIATIONS

Formalisme graphique :





NOTION DE PROPRIETES

PROPRIETES

Définition : Une propriété est une donnée élémentaire que l'on perçoit sur une entité ou une association.

Elle prend ses valeurs dans un domaine de valeurs de type simple (caractère, chaîne de caractères, entier, date).

PROPRIETES

Exemples :

Pour les entités :

- Nom, prénom pour l'entité ACTEURS
- Titre et Metteur en scène pou FILMS
- Nom et Adresse Pour CINEMA

PROPRIETES

Formalisme graphique : Pour les entités

FILMS
Titre
Metteur_en _scene

CINEMA
Nom
Adresse

ACTEURS
Nom
Prénom

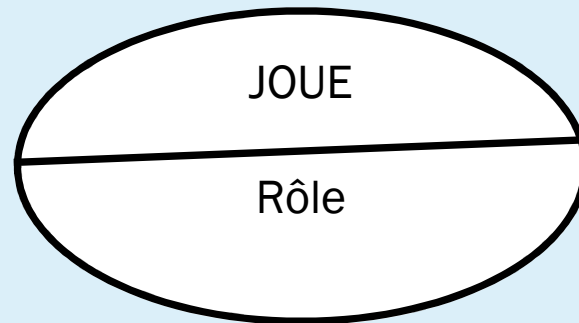
PROPRIETES

Formalisme graphique : Pour les associations :

Les propriétés à mettre dans une association doivent obligatoirement relier les entités.

PROPRIETES

Exemple: Rôle d'un acteur dans un film:
Le rôle relie un acteur et le film dans lequel il a joué.



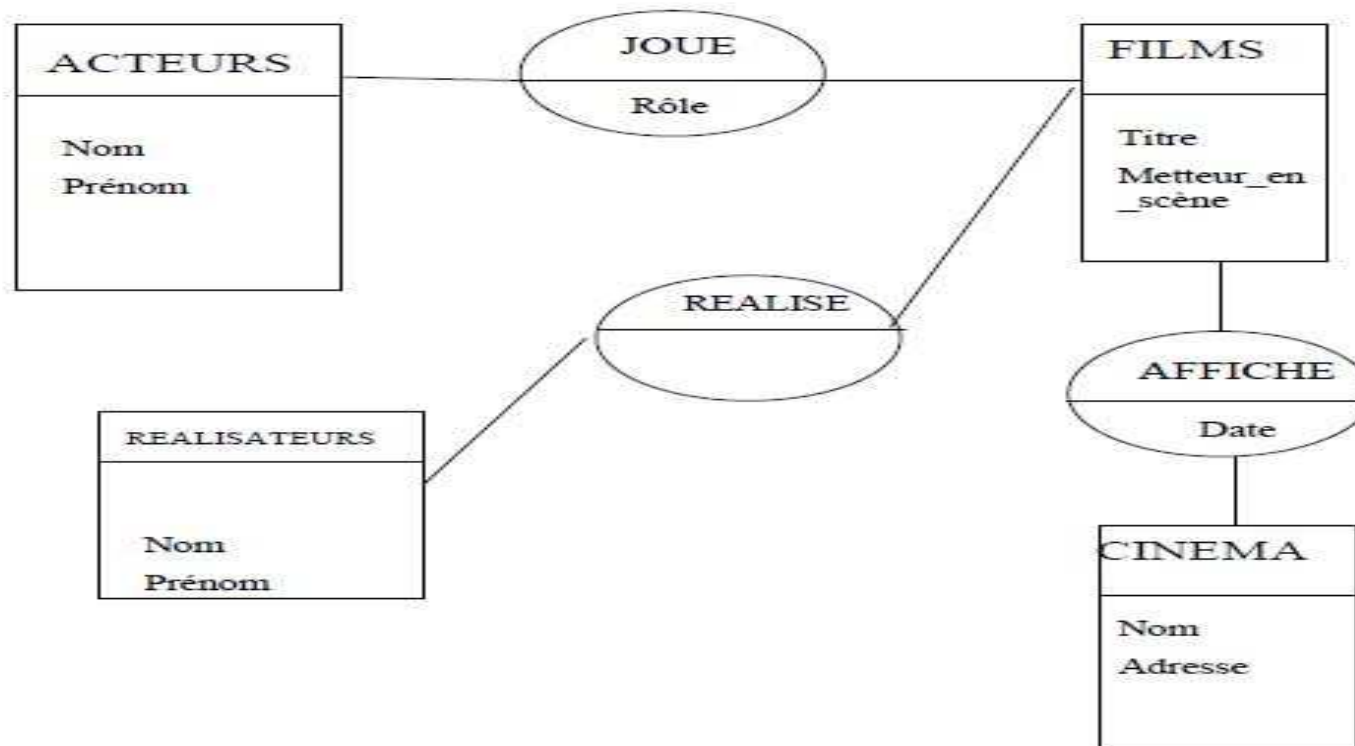
PROPRIETES

Autres exemples :

- L'entité `Personne` a pour propriétés :
`dateDeNaissance`, `nationalite`, `profession`.
- L'entité `Voiture` a pour propriétés : `Couleur`,
`kilometrage`.
- L'association `Possede` a pour propriétés :
`DateAchat`, `MontantAchat`.

SCHEMA GLOBAL

— Schéma global



EXERCICE 1 :

On se propose de représenter le système d'information d'une bibliothèque :

La bibliothèque enregistre chaque lecteur à qui elle donne un numéro de lecteur. Elle lui prend son nom et son adresse. Le lecteur peut éventuellement être membre d'une société adhérente. On enregistre alors l'identification de cette société.

Un lecteur peut emprunter plusieurs livres chaque jour. A chaque prêt, on associe une «date de retour au plus tard ».

Un lecteur appartient à un « type de lecteur ». Ce type lui permet d'avoir ou non accès à certaines catégories de livres.

La durée du prêt dépend de la catégorie du livre et du type de lecteur. Elle est la même pour tous les livres d'une catégorie donnée empruntés par un quelconque lecteur d'un type donné.

Un livre est caractérisé par son numéro d'inventaire. Il est nécessaire de connaître sa catégorie, le nom de son auteur, son éditeur, ainsi que le nombre de ses différents exemplaires disponibles. L'édition, lorsqu'elle existe, est également à connaître.

La catégorie d'un livre se repère par un numéro et possède un libellé. Il en est de même pour le type de lecteur.

Une société adhérente possède un nom et une adresse ; elle s'engage à envoyer un minimum de 500 lecteurs.

Donner le schéma entité-association correspondant à cet énoncé.

EXERCICE 2 :

Une entreprise veut améliorer sa gestion du matériel ainsi que celle des commandes de ces clients. Pour cela, elle envisage les orientations suivantes :

Elle veut connaître à tout instant la quantité disponible d'un matériel dans un magasin donné.

Les matériaux sont classés en catégories pour faciliter leur gestion. On doit pouvoir connaître les composants d'un matériel et les matériaux dans lesquels on trouve un composant donné. Lors d'une rupture de stock, un matériel peut être remplacé par un matériel de substitution.

Chaque client a un seuil maximal de commandes autorisé (droit d'approvisionnement) par catégorie de matériel pour une période donnée. Un client ne peut s'approvisionner que dans un magasin et un seul.

Une commande est définie par un numéro : elle concerne un seul client et différents matériaux et précise la quantité commandée.

Donner le schéma entité-association correspondant à cet énoncé.



LES IDENTIFIANTS

IDENTIFIANTS

DEFINITION:

Propriété ou groupe de propriétés dont la valeur identifie sans ambiguïté une entité ou une liaison d'une classe

Langage du E/A : identifiant=clé primaire.

Chaque occurrence doit pouvoir être repérée de manière unique et sans ambiguïté, pour être distinguée de toutes les autres.

Exemple : le numéro de matricule des employés

IDENTIFIANTS

REGLES POUR LES ENTITES:

- On souligne l'identifiant dans la représentation graphique;
- Chaque entité doit avoir un seul identifiant.
- On peut aussi au besoin créer un identifiant artificiel par commodité pour une entité.

IDENTIFIANTS

REGLES POUR LES ENTITES:

Exemple:

FILMS
<u>NumFilm</u>
Titre
Metteur_en_scene

ACTEURS
<u>NumActeur</u>
Nom
Prénom

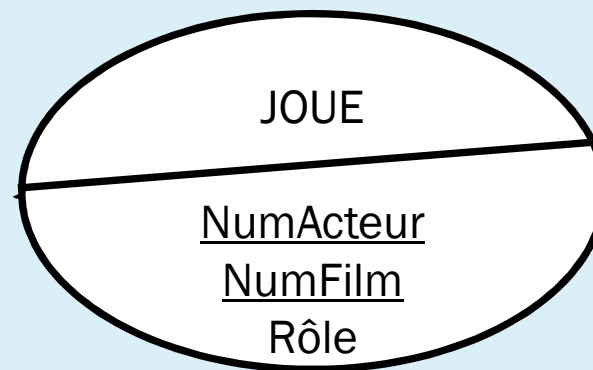
IDENTIFIANTS

REGLE POUR LES ASSOCIATIONS:

L'identifiant pour une association est obtenu en juxtaposant les identifiants des entités qui participent à la liaison.

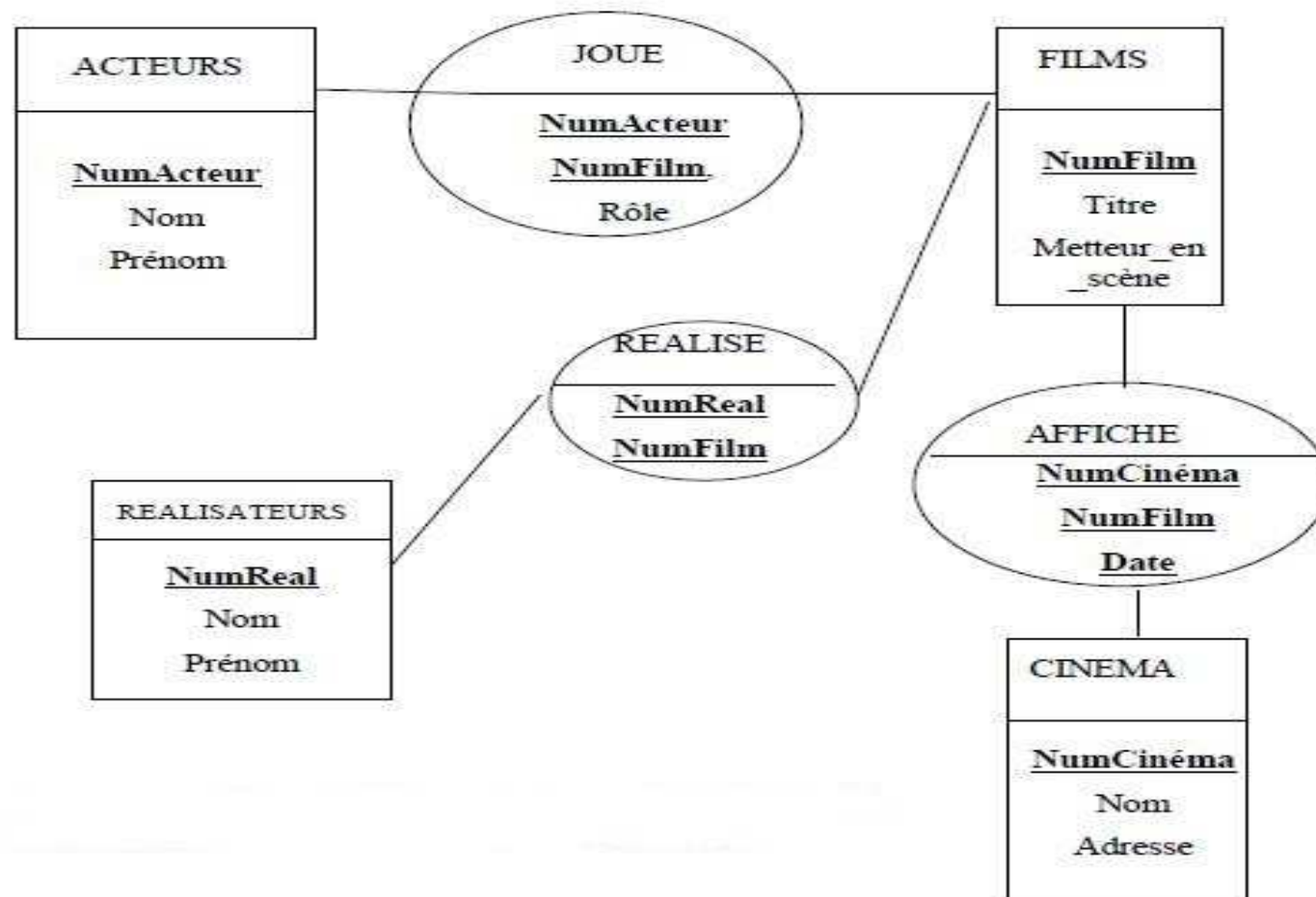
Exemple:

un acteur joue un Film.



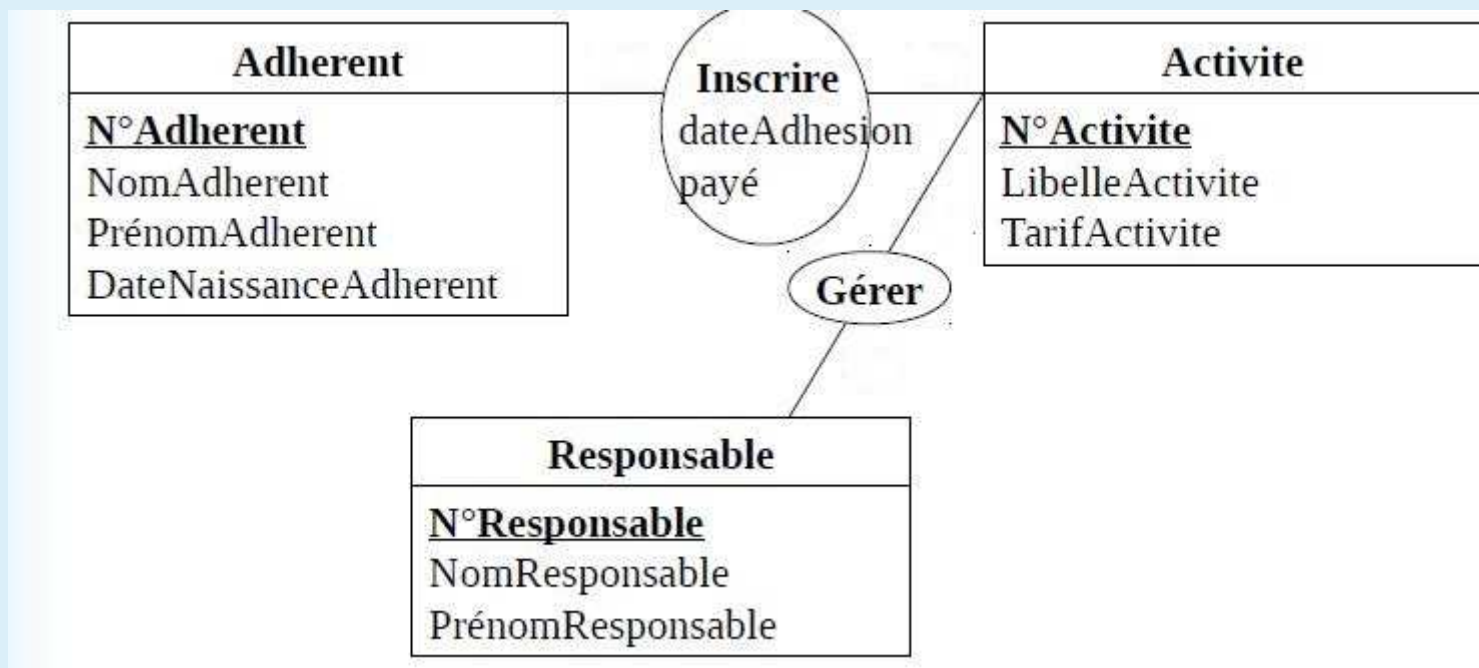
IDENTIFIANTS

SCHEMA GLOBAL:



EXERCICE 2 :

Corrigé:



EXERCICE 1 :

On se propose de représenter le système d'information d'une bibliothèque :

La bibliothèque enregistre chaque lecteur à qui elle donne un numéro de lecteur. Elle lui prend son nom et son adresse. Le lecteur peut éventuellement être membre d'une société adhérente. On enregistre alors l'identification de cette société.

Un lecteur peut emprunter plusieurs livres chaque jour. A chaque prêt, on associe une «date de retour au plus tard ».

Un lecteur appartient à un « type de lecteur ». Ce type lui permet d'avoir ou non accès à certaines catégories de livres.

La durée du prêt dépend de la catégorie du livre et du type de lecteur. Elle est la même pour tous les livres d'une catégorie donnée empruntés par un quelconque lecteur d'un type donné.

Un livre est caractérisé par son numéro d'inventaire. Il est nécessaire de connaître sa catégorie, le nom de son auteur, son éditeur, ainsi que le nombre de ses différents exemplaires disponibles. L'édition, lorsqu'elle existe, est également à connaître.

La catégorie d'un livre se repère par un numéro et possède un libellé. Il en est de même pour le type de lecteur.

Une société adhérente possède un nom et une adresse ; elle s'engage à envoyer un minimum de 500 lecteurs.

Donner le schéma entité-association correspondant à cet énoncé.

EXERCICE 2 :

Une association gère des adhérents (nom, prénom, date de naissance) qui s'inscrivent à un choix d'activités (Bridge, Tricot, Judo...).

Un tarif de cotisation annuelle est fixé pour chaque activité.

Chaque activité a un seul responsable (nom, prénom) et un nombre de participants maximum.

Pour chaque activité, on indique la date d'adhésion du participant et s'il a payé sa cotisation.

Donner le schéma entité-association correspondant à cet énoncé.



LES CARDINALITES

LES CARDINALITES

DEFINITION:

La cardinalité permet d'exprimer la façon dont sont associées les entités. Elle définit les restrictions sur le nombre d'associations possibles entre deux entités.

Exemple :

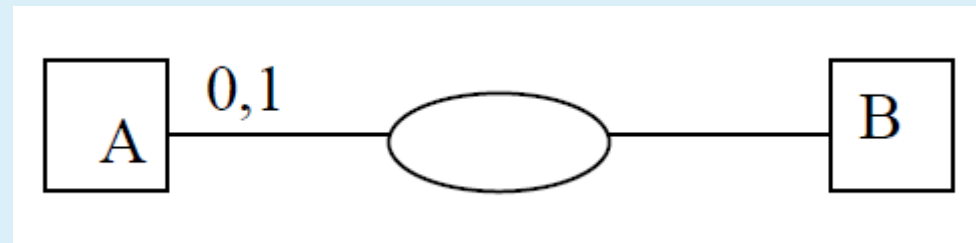
- Une voiture n'est possédée que par une personne.
- Une personne peut posséder plusieurs voitures.

LES CARDINALITES

TYPES DE CARDINALITE:

TYPE 0,1 : une entité de A peut être reliée à aucune ou à une seule entité de B.

Formalisme:

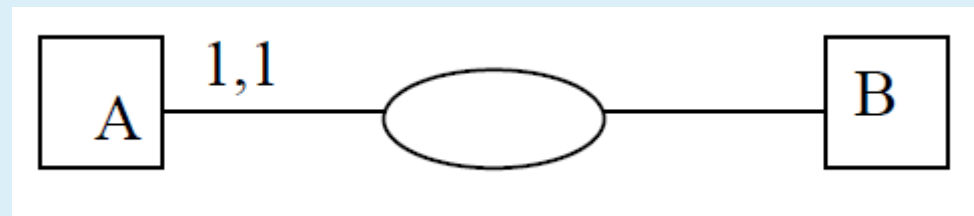


LES CARDINALITES

TYPES DE CARDINALITE:

TYPE 1,1 : une entité de A est reliée à une seule entité de B.

Formalisme:

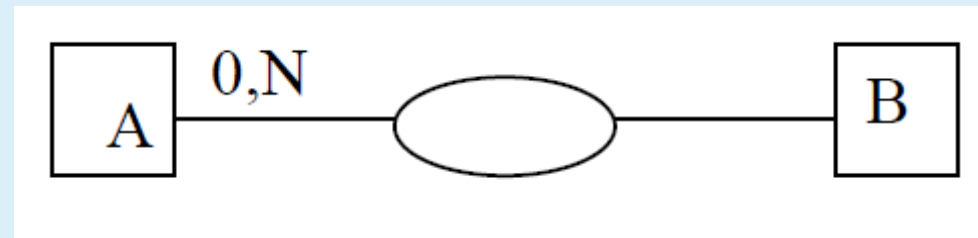


LES CARDINALITES

TYPES DE CARDINALITE:

TYPE de 0 à plusieurs (0,N) : une entité de A peut être reliée à aucune ou à plusieurs entités de B

Formalisme:

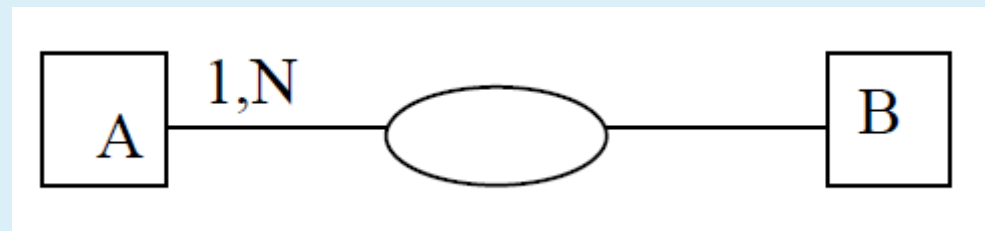


LES CARDINALITES

TYPES DE CARDINALITE:

TYPE de 1 à plusieurs (1,N) : une entité de A peut être reliée à une ou plusieurs entités de B

Formalisme:



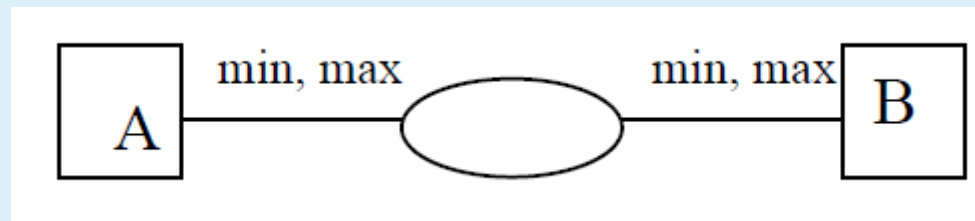
LES CARDINALITES

TYPES DE CARDINALITE:

La cardinalité est donc défini par deux nombres (min,max) représentant le nombre de fois minimum et le nombre de fois maximum qu'une entité participe à une association.

Les valeurs possibles sont : (0,1), (1,1); (0,N), (1,N).

Formalisme:

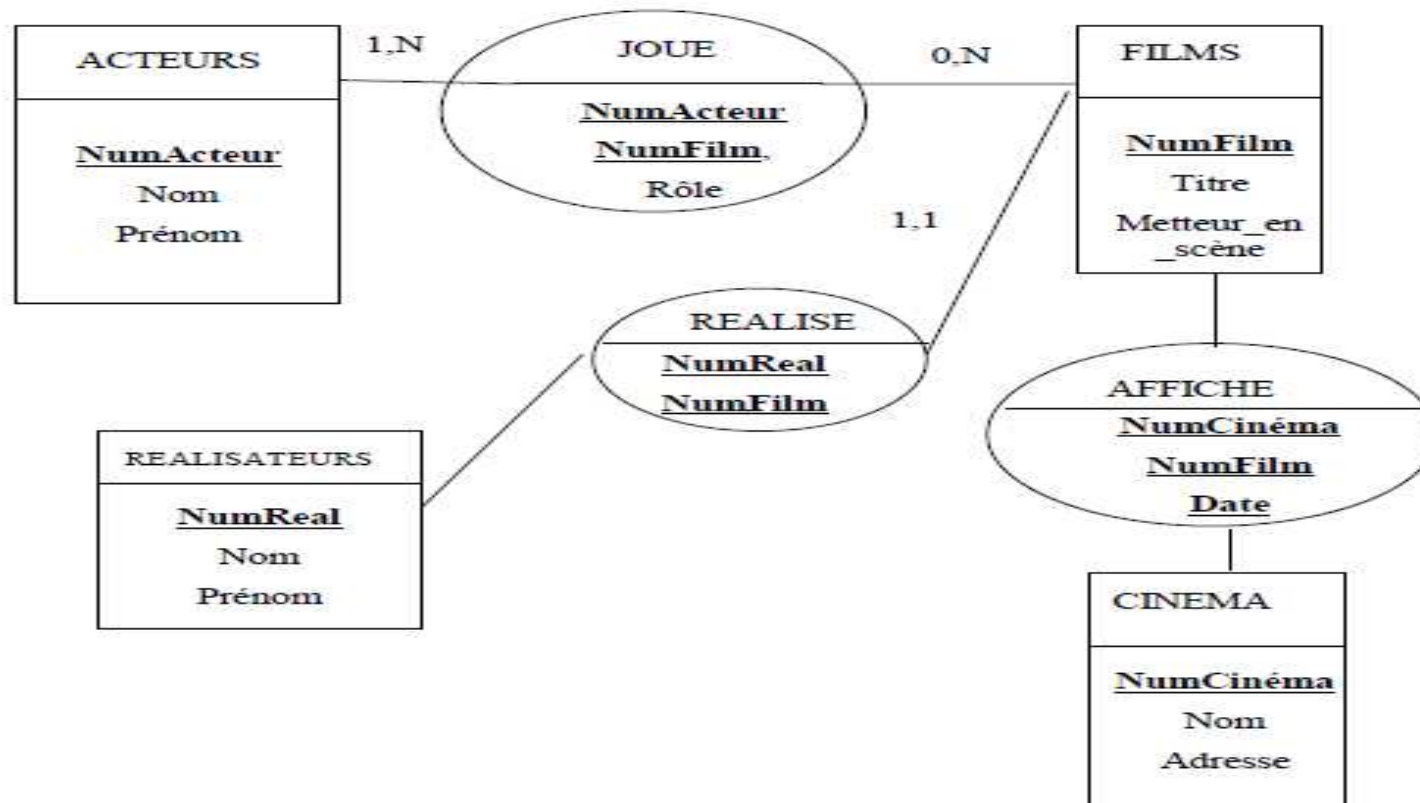


DEMARCHE DE CONCEPTION

- Identifier les entités les plus naturelles (sujets ou compléments)
- Identifier les associations entre ces entités (verbe qui définit la relation entre entités);
- Identifier les attributs et les identifiants de ces entités et de ces associations.
- Déterminer les cardinalités en distinguant le minimum et le maximum : les contraintes d'intégrité.

LES CARDINALITES

Formalisme global pour l'exemple sur les acteurs:



SUITE EXERCICE 1 :

On se propose de représenter le système d'information d'une bibliothèque :

La bibliothèque enregistre chaque lecteur à qui elle donne un numéro de lecteur. Elle lui prend son nom et son adresse. Le lecteur peut éventuellement être membre d'une société adhérente. On enregistre alors l'identification de cette société.

Un lecteur peut emprunter plusieurs livres chaque jour. A chaque prêt, on associe une «date de retour au plus tard ».

Un lecteur appartient à un « type de lecteur ». Ce type lui permet d'avoir ou non accès à certaines catégories de livres.

La durée du prêt dépend de la catégorie du livre et du type de lecteur. Elle est la même pour tous les livres d'une catégorie donnée empruntés par un quelconque lecteur d'un type donné.

Un livre est caractérisé par son numéro d'inventaire. Il est nécessaire de connaître sa catégorie, le nom de son auteur, son éditeur, ainsi que le nombre de ses différents exemplaires disponibles. L'édition, lorsqu'elle existe, est également à connaître.

La catégorie d'un livre se repère par un numéro et possède un libellé. Il en est de même pour le type de lecteur.

Une société adhérente possède un nom et une adresse ; elle s'engage à envoyer un minimum de 500 lecteurs.

Donner le schéma entité-association correspondant à cet énoncé.



MODÈLE RELATIONNEL

MODELE RELATIONNEL

ETAPES CONCEPTION DE BASE DE DONNEES (1/3):

On distingue quatre étapes dans la conception d'une base de données en modèle relationnel :

- **L'analyse** : Consiste à étudier le problème et à consigner dans un document : la note de clarification, les besoins, les choix, les contraintes.
- **La modélisation conceptuelle** : Décrire le problème posé, de façon non-formelle (en générale graphique), en prenant des hypothèses de simplification. C'est une représentation simplifiée d'une réalité.

MODELE RELATIONEL

ETAPES CONCEPTION DE BASE DE DONNEES (2/3):

- **La modélisation logique** : Décrire la solution, en prenant une orientation informatique générale (type de SGBD), formelle, mais indépendamment de choix d'implémentation spécifiques;
- **L'implémentation** : Elle correspond aux choix techniques, en terme de SGBD choisi et à leur mise en œuvre (programmation, optimisation...).

MODELE RELATIONEL

ETAPES CONCEPTION DE BASE DE DONNEES (3/3):

Le modèle EA fait intervenir les notions suivantes :
Entité, Association, Propriété, Identifiant,
cardinalité;

Le modèle relation aussi fait intervenir les notions
suivantes:

- Table ou relation;
- Attributs;
- Clé candidate d'une relation.

MODELE RELATIONEL

NOTION DE TABLES ou RELATIONS (1/2):

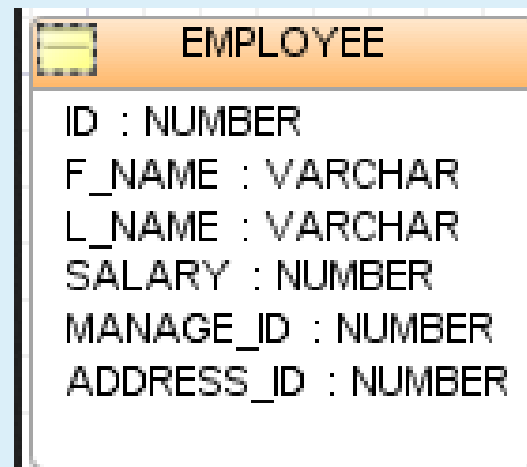
Une **table** est un ensemble de données organisées sous forme d'un tableau où les colonnes correspondent à **des catégories d'information** et les lignes à des **enregistrements**, également appelés entrées ou **occurrences**.

Les relations ou tables représentent **les objets du monde réel** (comme les personnes, les animaux, les objets de la nature, etc.) ou **les associations** entre ces objets.

MODELE RELATIONNEL

NOTION DE TABLES ou RELATIONS (2/2): :

Exemple: Une table « Employé » qui stocke le nom, prénom, salaire, etc.



The diagram shows a table structure for 'EMPLOYEE'. It has a yellow header bar with the table name and a list of attributes with their data types below it.

EMPLOYEE
ID : NUMBER
F_NAME : VARCHAR
L_NAME : VARCHAR
SALARY : NUMBER
MANAGE_ID : NUMBER
ADDRESS_ID : NUMBER

MODELE RELATIONEL

REPRESENTATION D'UNE RELATION:

une relation (entité ou association) est représentée par:

- **nom_de_l'entité** (liste des attributs de l'entité);
- **nom_de_l'association** (liste des identifiants des entités participantes, liste des attributs de l'association);

Exemple: CLIENT(IdCli, nom, ville)

MODELE RELATIONNEL

NOTION D'ATTRIBUTS:

Chaque composante d'une relation est un attribut. Ce sont **les colonnes** qui contiennent un ensemble de valeurs d'un domaine.

Exemple : représentation tabulaire d'une relation:

SALARIE			
Matricule	Nom	Grade	Salaire
100	Müller	cadre	12'000
101	Roche	employé	4'500
102	Chapuis	assistant	4'000

⇒ Nom table

⇒ Attributs

⇒ Tuples

MODELE RELATIONEL

NOTION DE CLES D'UNE RELATION:

clé candidate - Une clé candidate d'une relation est un ensemble minimal des attributs de la relation dont les valeurs identifient à coup sûr un objet.

clé primaire - La clé primaire d'une relation est une de ses clés candidates. Pour signaler la clé primaire, ses attributs sont généralement soulignés.

clé étrangère - Une clé étrangère dans une relation est formée d'un ou plusieurs attributs qui constituent une clé primaire dans une autre relation.



PASSAGE DU MODÈLE ENTITÉ/ASSOCIATION AU MODÈLE RELATIONNEL

**PASSAGE DU MODÈLE ENTITÉ/ASSOCIATION AU
MODÈLE RELATIONNEL**

COMPARAISON DES DEUX MODÈLES:

Modèle EA	Modèle Relationnel
Association; Entité	Table(relation)
Propriété	Attribut
Identifiant	Clé Primaire

PASSAGE DU MODÈLE ENTITÉ/ASSOCIATION AU MODÈLE RELATIONNEL

ETAPES de PASSAGE MEA vers MR:

Etape 1 : Toute entité du modèle entité/association est représentée par une **relation** dans le schéma relationnel équivalent. La clé de cette relation est l'identifiant de la classe d'entités correspondante.

Exemple:

ACTEURS
<u>NumActeur</u>
Nom
Prénom

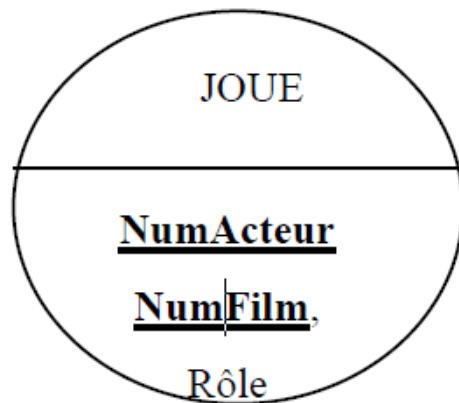
ACTEURS(NumActeur, Nom, Prénom)

PASSAGE DU MODÈLE ENTITÉ/ASSOCIATION AU MODÈLE RELATIONNEL

ETAPES de PASSAGE MEA vers MR:

Etape 2 : Toute association est transformée en **relation**. La clé de cette relation est composée de tous les identifiants des entités participantes.

Exemple:



JOUER(NumActeur, NumFilm, Rôle)

PASSAGE DU MODÈLE ENTITÉ/ASSOCIATION AU MODÈLE RELATIONNEL

ETAPES de PASSAGE MEA vers MR:

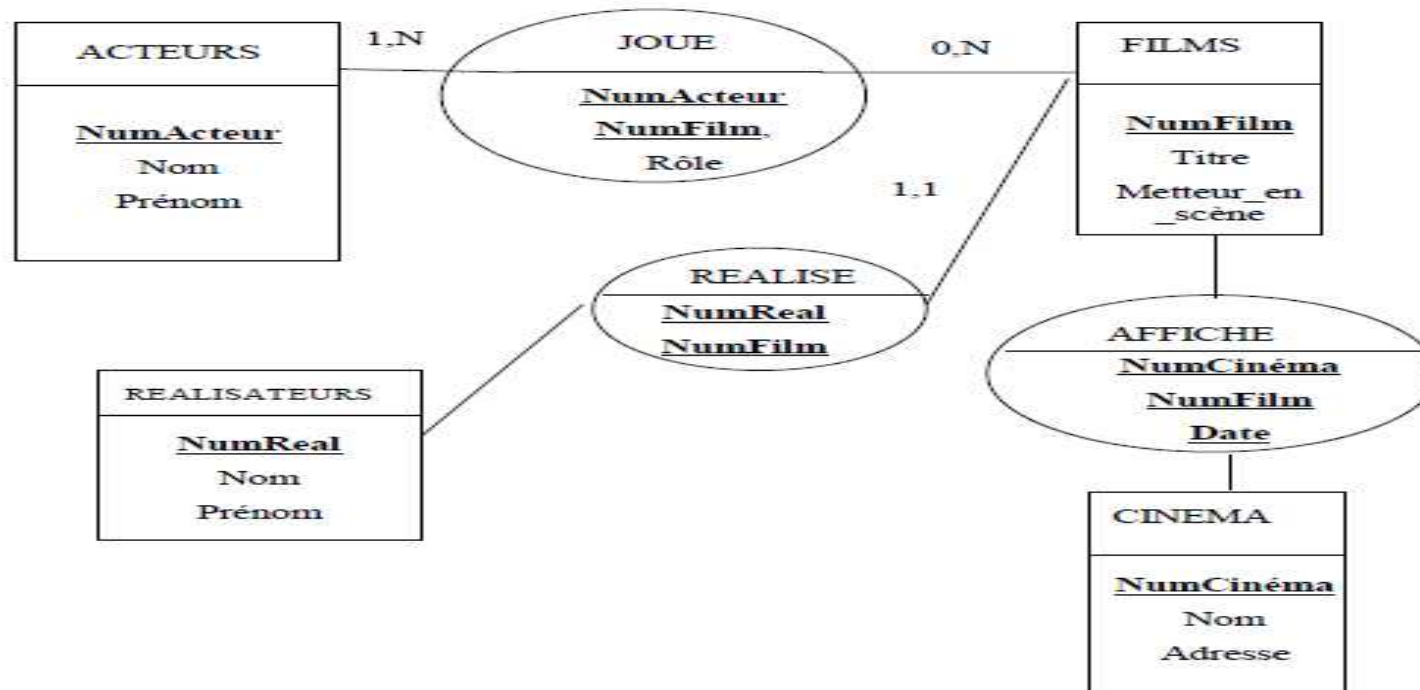
Etape 3 : Toute association reliée à une entité avec une cardinalité de **type 0,1 ou 1,1** peut être fusionnée avec cette entité.

Dans ce cas on déplace les attributs de l'association vers ceux de la relation traduisant l'entités.

PASSAGE DU MODÈLE ENTITÉ/ASSOCIATION AU MODÈLE RELATIONNEL

ETAPES de PASSAGE MEA vers MR:

Le schéma peut être optimisé car il contient une association de type 1,1.



PASSAGE DU MODÈLE ENTITÉ/ASSOCIATION AU MODÈLE RELATIONNEL

ETAPES de PASSAGE MEA vers MR:

A l'issue de l'étape de 2 de la transformation, on devrait obtenir ce qui suit:

- ACTEURS(NumActeur, Nom, Prénom)
- FILMS(NumFilm, Titre, MetteurEnScène)
- CINEMA(NumCinéma, Nom, Adresse)
- REALISATEURS(NumReal, Nom, Prénom)
- JOUE(NumA^Cteur, NumFilm, Rôle)
- AFFICHE(NumFilm, NumCinema, Date)
- REALISE(NumFilm, NumRéal)

PASSAGE DU MODÈLE ENTITÉ/ASSOCIATION AU MODÈLE RELATIONNEL

ETAPES de PASSAGE MEA vers MR:

Selon la règle d'optimisation énoncée à l'étape 3, on obtient ce qui suit.

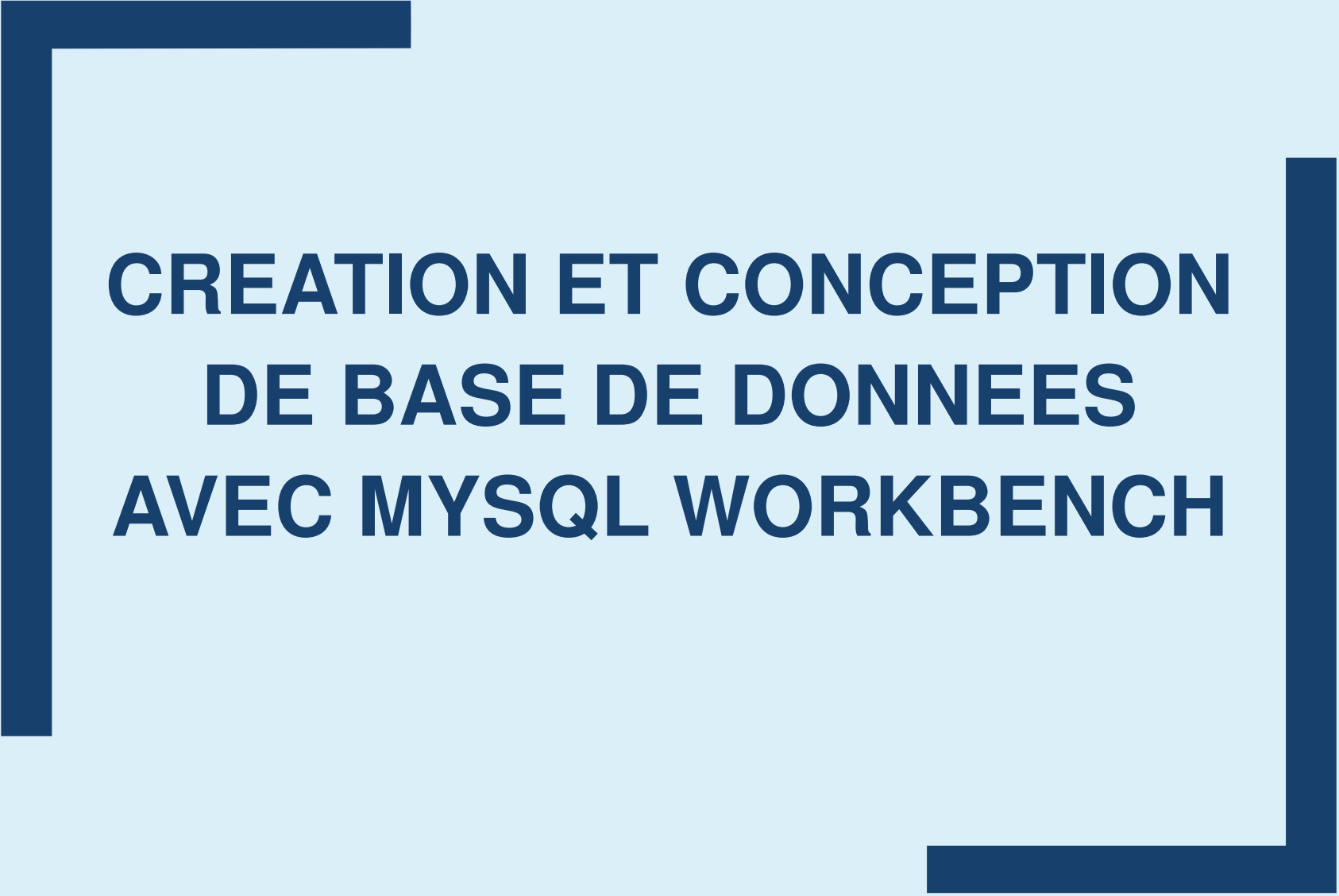
- La table qui traduit l'association REALISE doit disparaître;
- On déplace les attributs de REALISE (NumFilm et NumReal) vers ceux de FILMS (côté duquel on a (1,1));
- La table Films devient alors :

FILMS(**NumFilm**, Titre, MetteurEnScène, NumReal).

Ce qui donne la représentation suivante :

PASSAGE DU MODÈLE ENTITÉ/ASSOCIATION AU MODÈLE RELATIONNEL

- ACTEURS(NumActeur, Nom, Prénom)
- FILMS(NumFilm, Titre, MetteurEnScène, *NumReal*)
- CINEMA(NumCinéma, Nom, Adresse)
- REALISATEURS(NumReal, Nom, Prénom)
- JOUE(NumA*ct*eur, NumFilm, Rôle)
- AFFICHE(NumFilm, NumCinema, Date)

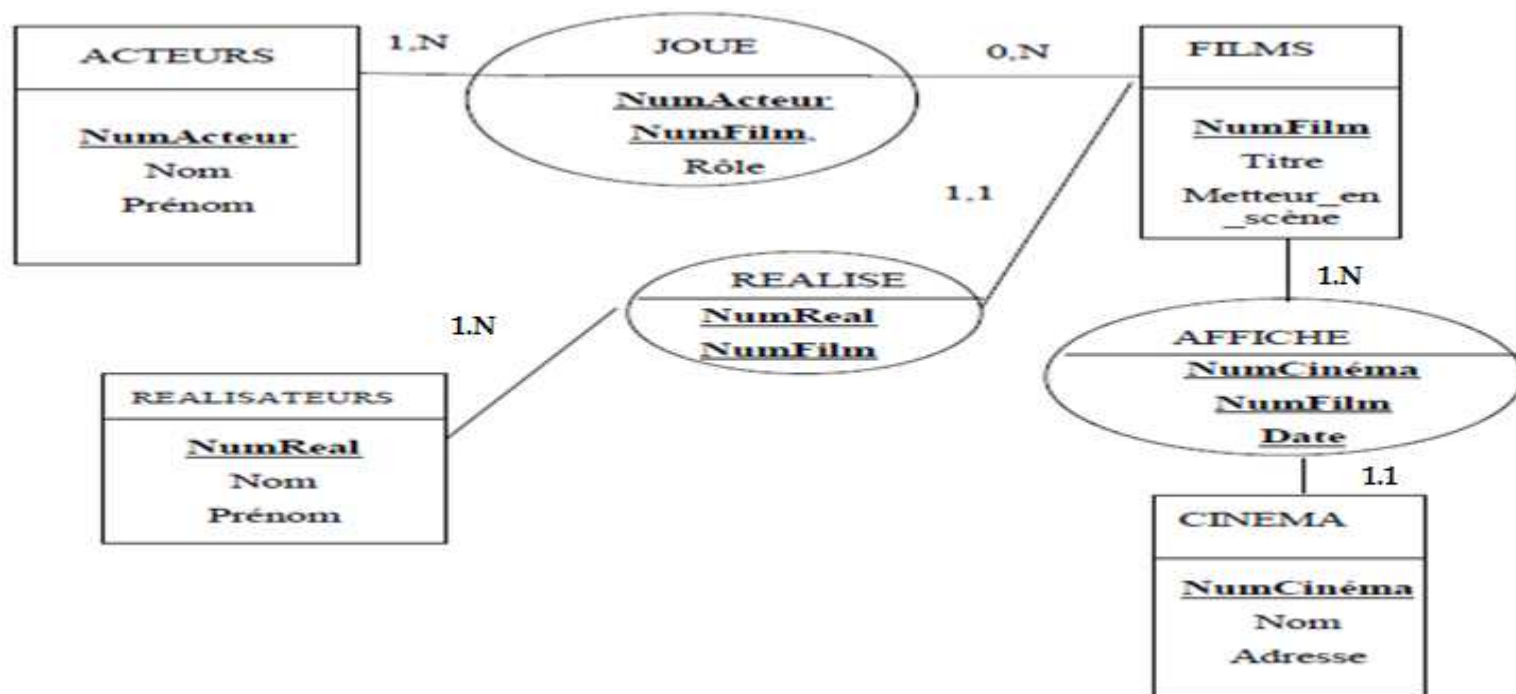


CREATION ET CONCEPTION DE BASE DE DONNEES AVEC MYSQL WORKBENCH

CREATION ET CONCEPTION DE BASE DE DONNEES AVEC MYSQL WORKBENCH

Introduction

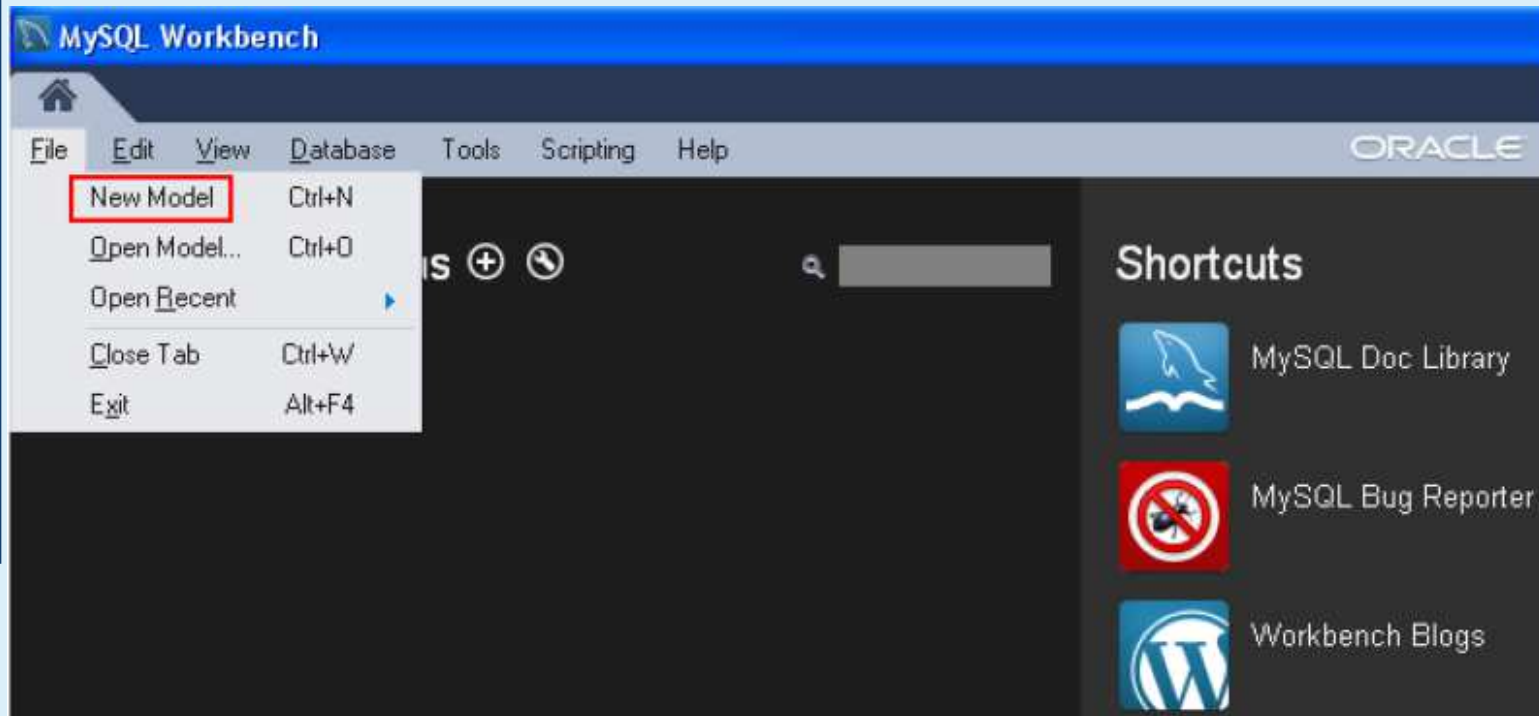
Conception de la base de données représentée sur le diagramme suivant:



CREATION ET CONCEPTION DE BASE DE DONNEES AVEC MYSQL WORKBENCH

Créer un modèle :

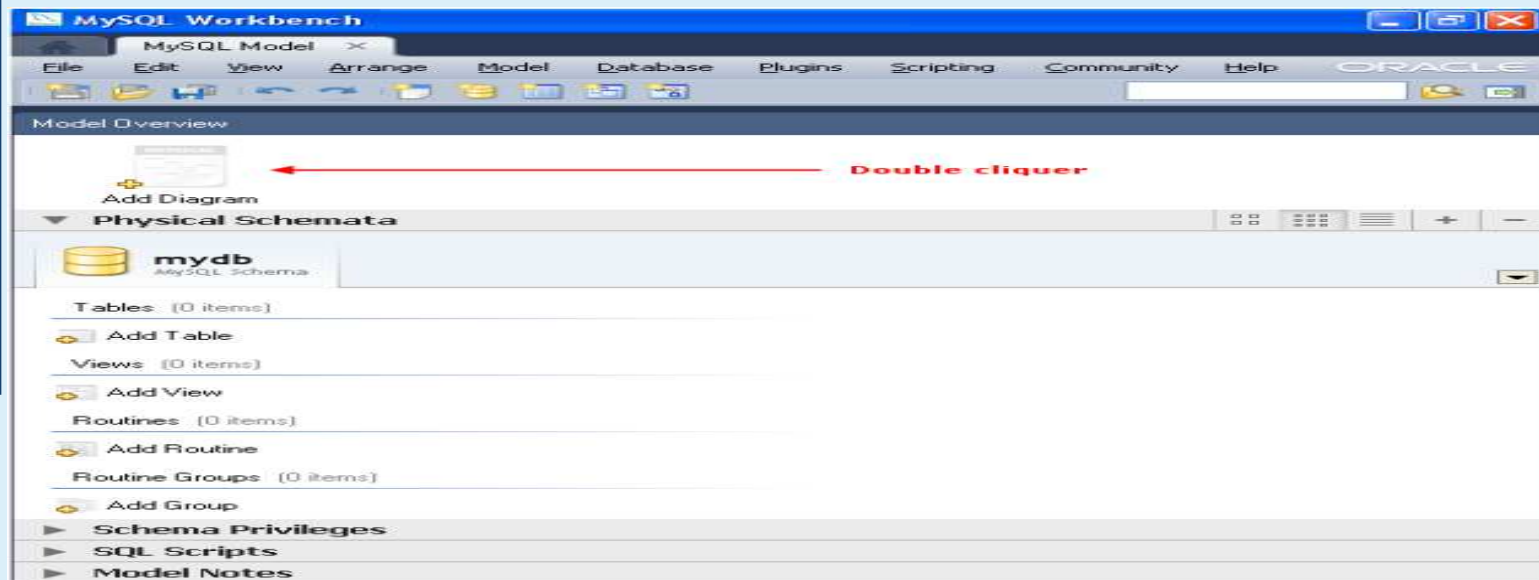
Onglet « File » puis « New Model »



CREATION ET CONCEPTION DE BASE DE DONNEES AVEC MYSQL WORKBENCH

Créer un modèle :

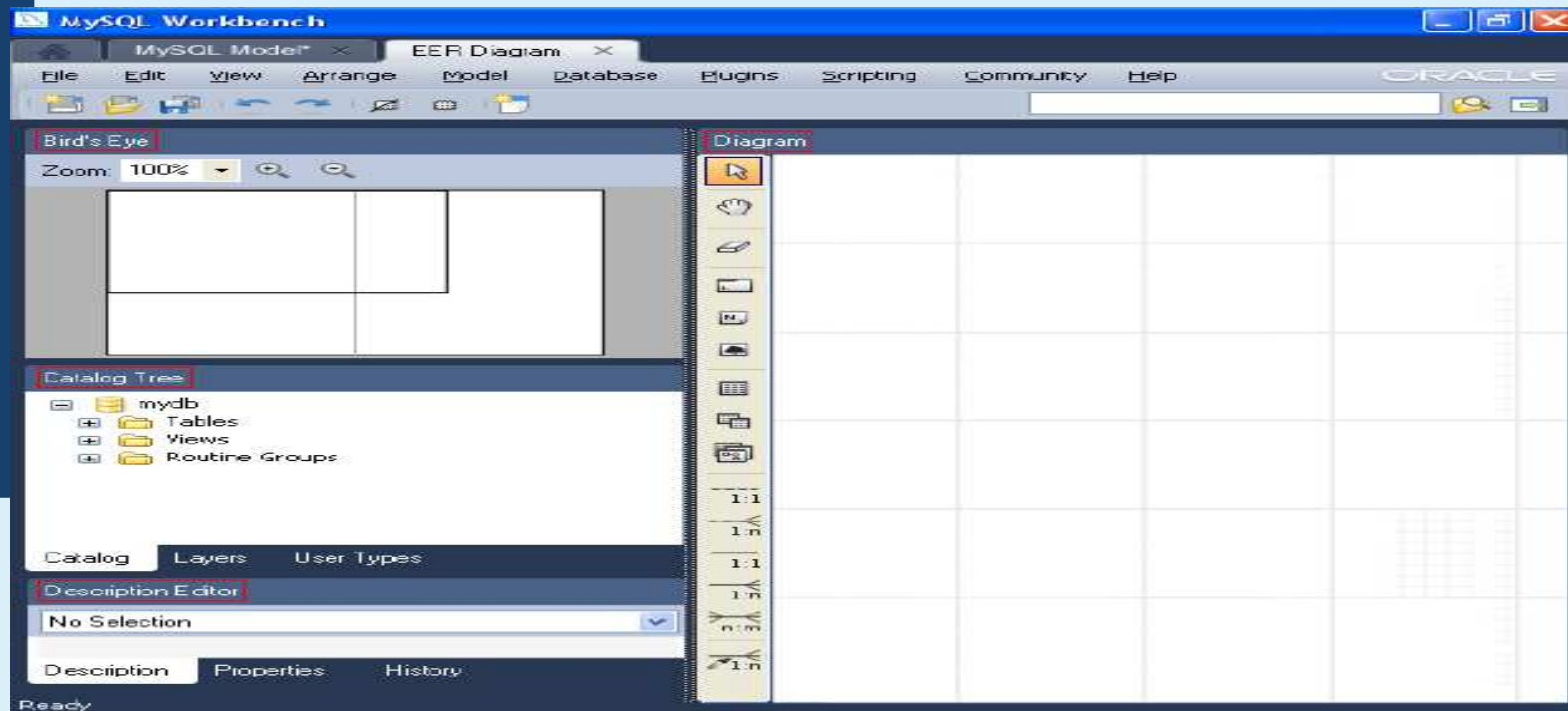
MySQL Workbench propose la fenêtre suivante permettant de créer un MLD (modèle logique de données) sous la forme d'un diagramme



CREATION ET CONCEPTION DE BASE DE DONNEES AVEC MYSQL WORKBENCH

Créer un modèle :

Double-clique sur « Add Diagram » pour passer à la création effective d'un diagramme



Créer un modèle :

L'espace de travail de MySQL Workbench :

- **Diagram**, notre espace principal de travail;
- **Bird's Eye** : permet au besoin de réduire la taille des objets;
- **Catalog Tree** : permet de représenter les objets sous forme arborescente;
- **Description Editor** : permet d'embellir les objets (couleur, etc.)
- **Modeling Additions** : structures de tables prédéfinies (*templates*) squelettiques;

Créer un modèle :

On peut commencer par choisir le nom du schéma (au sens SQL du terme), en remplaçant celui qui est fourni par défaut (« mydb »).

On remplace « mydb » par « Training_ITS1 ».

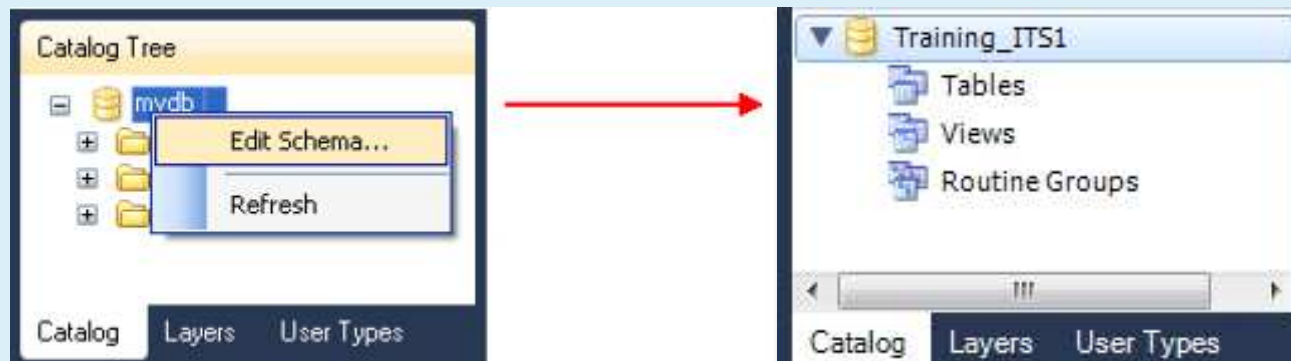
Procédure:

Clic droit sur « mydb » puis clic sur « Edit schema »).

Créer un modèle :

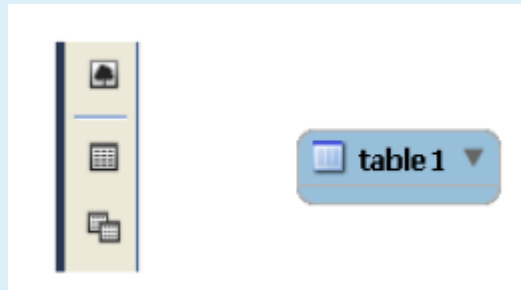
Procédure:

Clic droit sur « mydb » puis clic sur « Edit schema »).



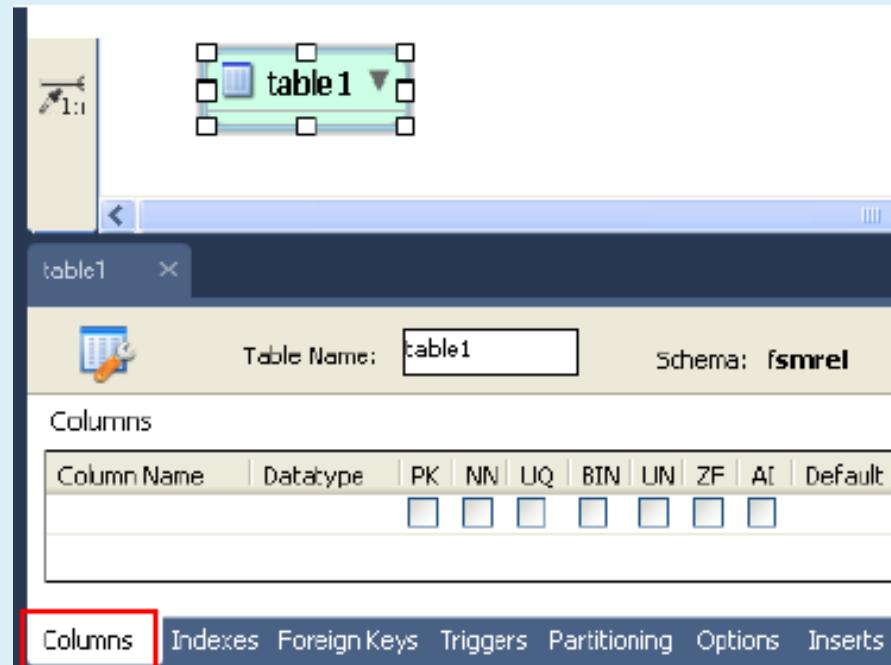
Définir la structure des tables

Cliquer sur l'icone « New table » et cliquer à nouveau dans l'espace vide:



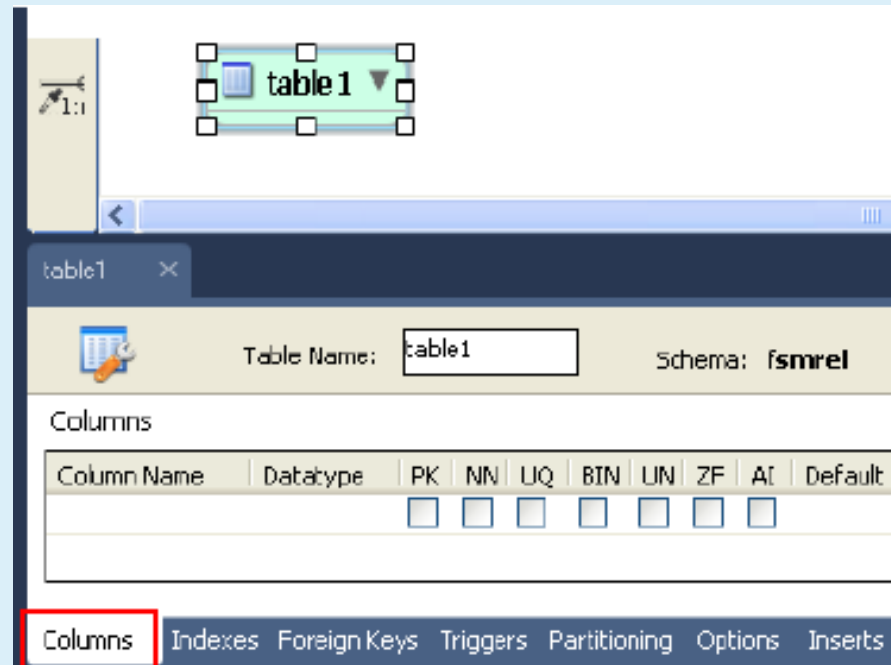
Définir la structure des tables

Renommer la table et passer à la construction de la structure de son en-tête (ensemble des colonnes de la table) cliquant sur l'objet «table1»



Définir la structure des tables

Renommer la table et passer à la construction de la structure de son en-tête (ensemble des colonnes de la table) cliquant sur l'objet «table1»



Définir la structure des tables

Datatype pour les attributs:

PK : La colonne appartient à la clé primaire;

NN : Not Null;

UQ : La colonne appartient à une clé étrangère;

BIN : Colonne binaire;

UN : Le type de la colonne est non signé;

ZF : Remplir la colonne (si numérique) avec des zéro;

AI : Auto_Incrément;

Default : valeur par défaut.

Définir la structure des tables

Datatype pour les attributs:

PK : La colonne appartient à la clé primaire;

NN : Not Null;

UQ : La colonne appartient à une clé étrangère;

BIN : Colonne binaire;

UN : Le type de la colonne est non signé;

ZF : Remplir la colonne (si numérique) avec des zéro;

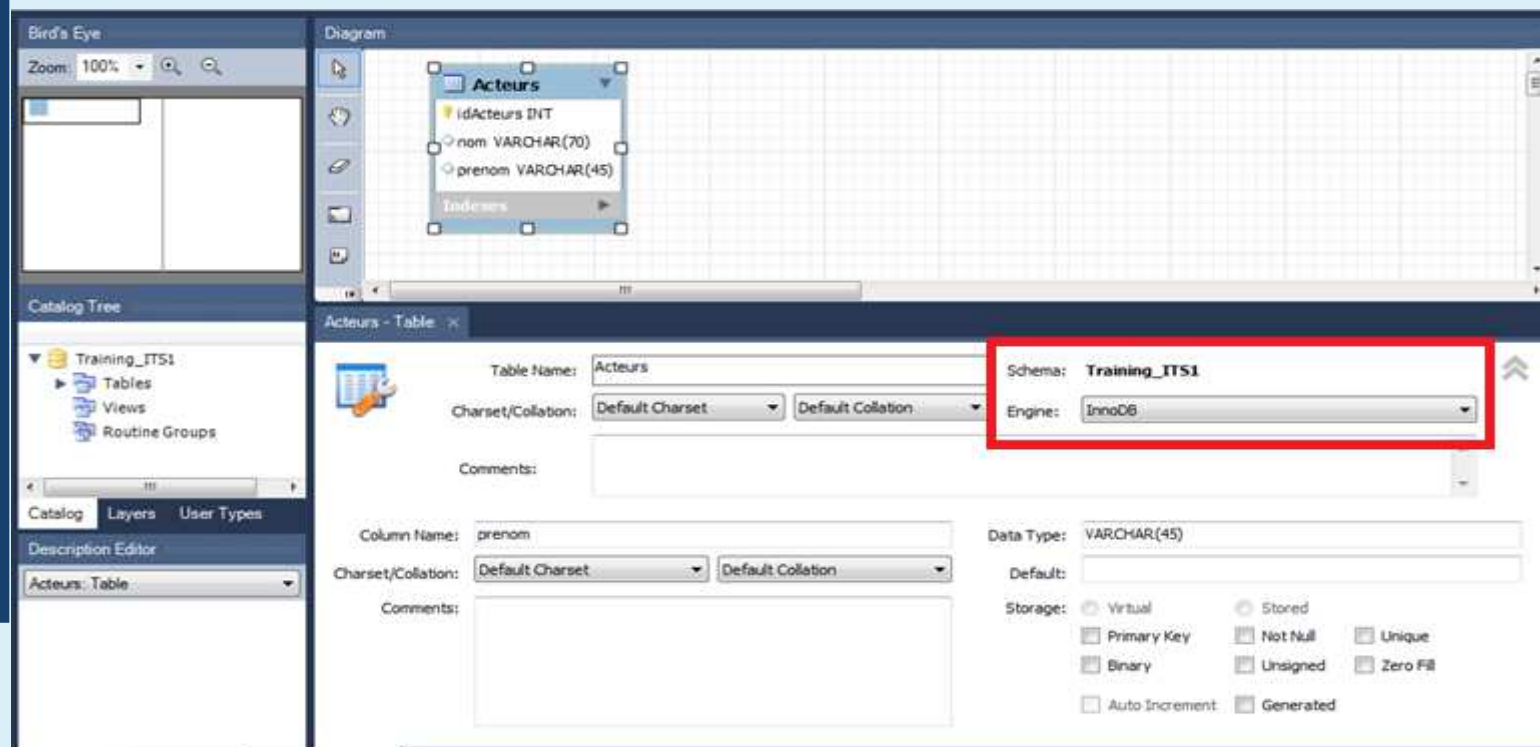
AI : Auto_Incrément;

Default : valeur par défaut.

CREATION ET CONCEPTION DE BASE DE DONNEES AVEC MYSQL WORKBENCH

Définir la structure des tables

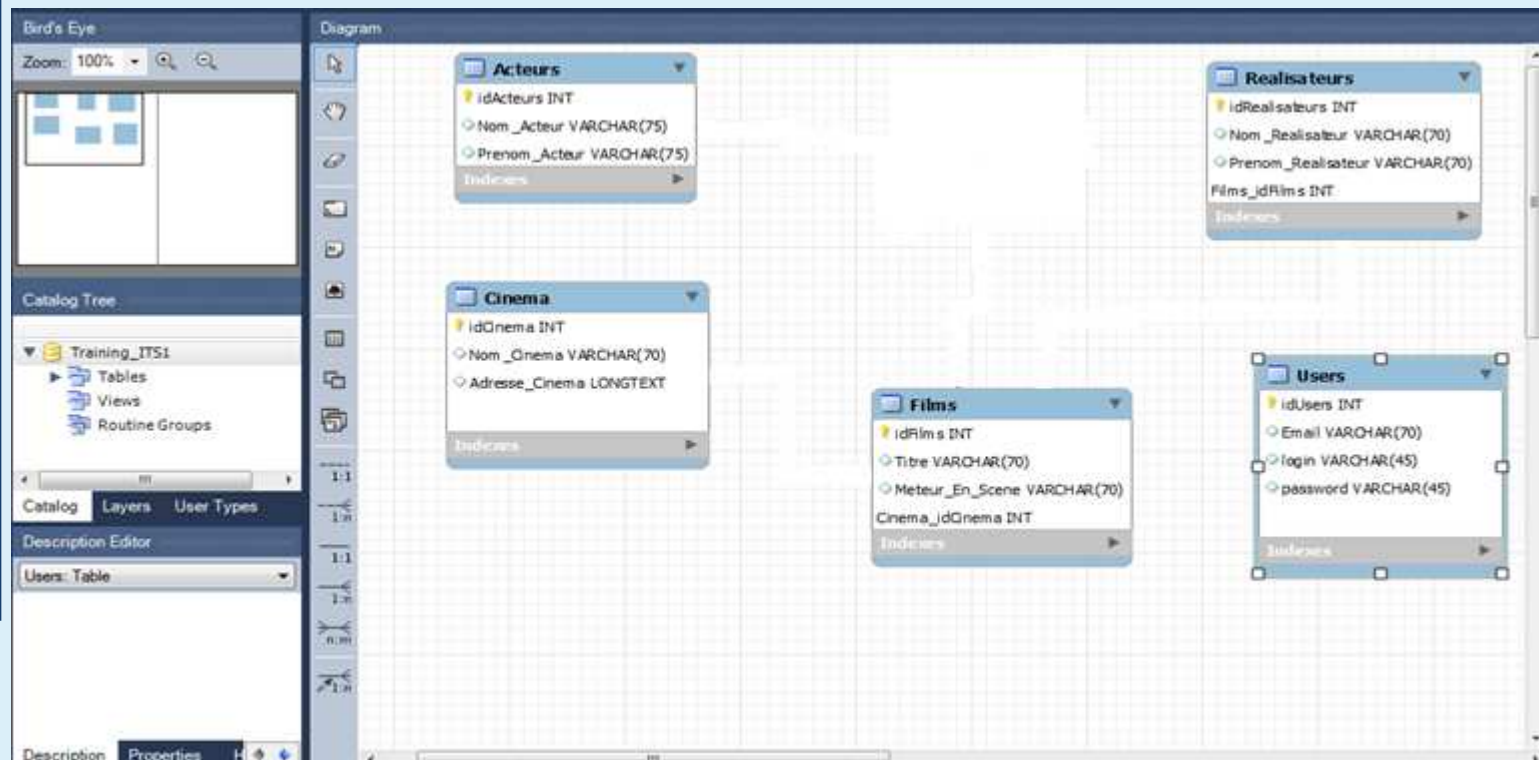
Ajout de la table « Acteurs » :



CREATION ET CONCEPTION DE BASE DE DONNEES AVEC MYSQL WORKBENCH

Définir la structure des tables

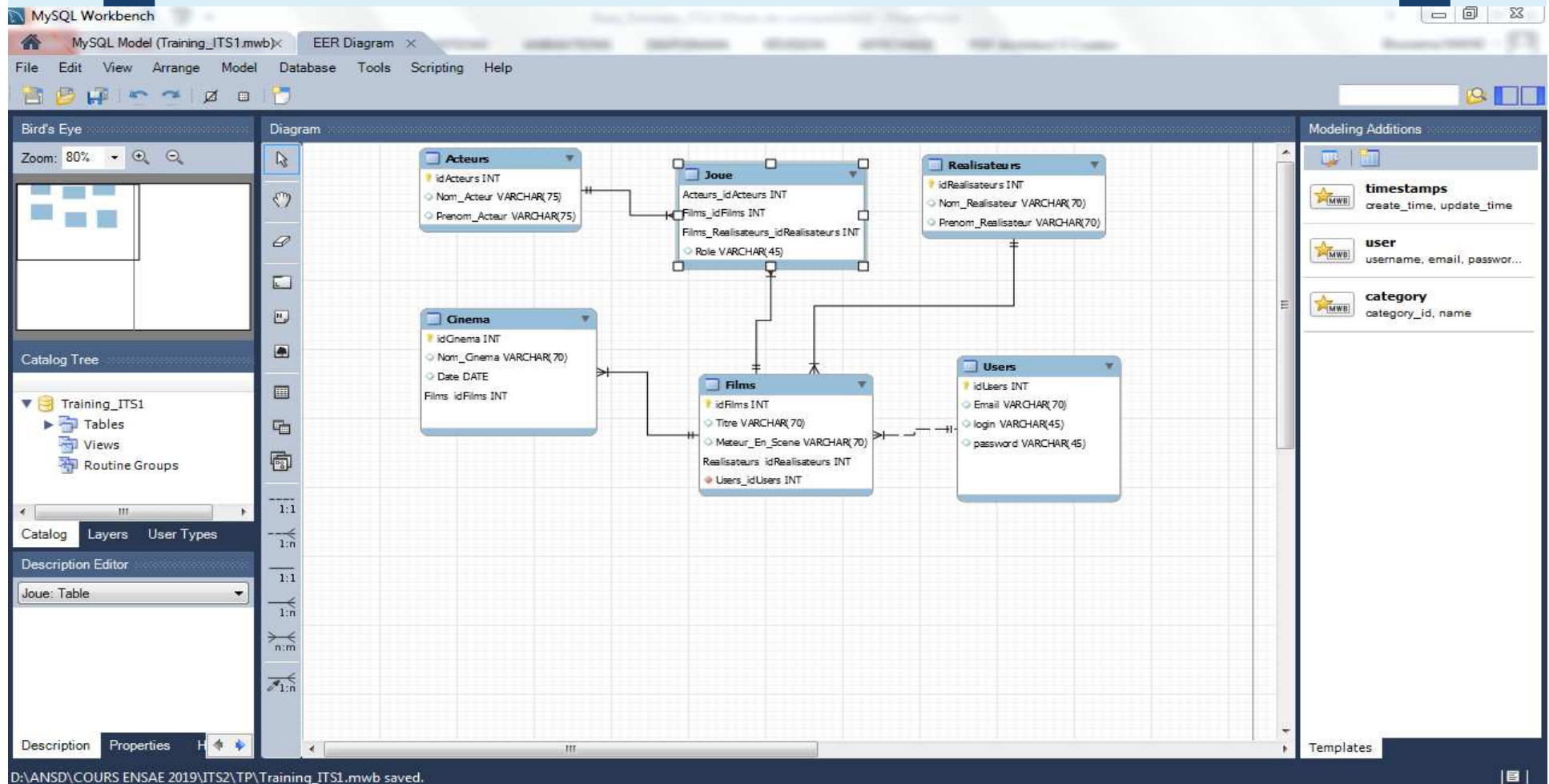
Ajout des autres tables :



CREATION ET CONCEPTION DE BASE DE DONNEES AVEC MYSQL WORKBENCH

Définir la structure des tables

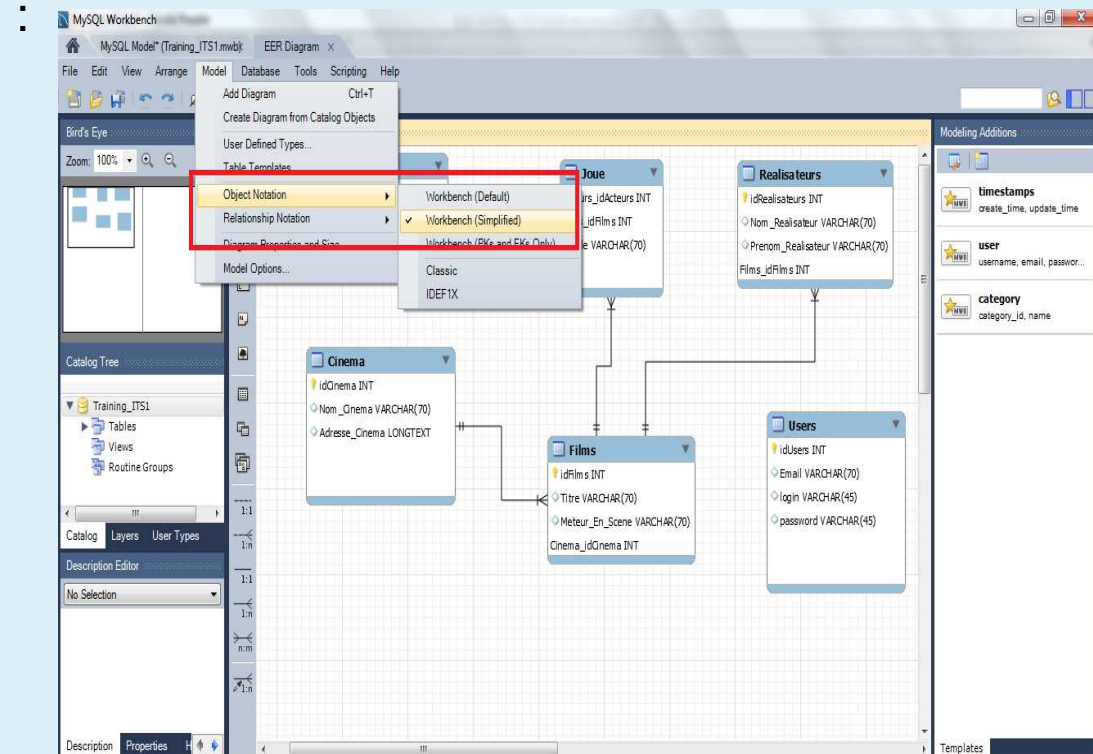
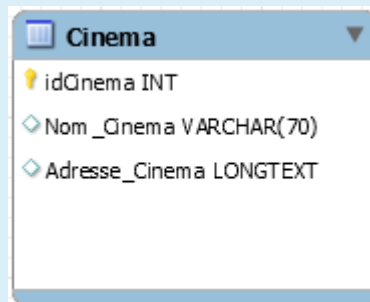
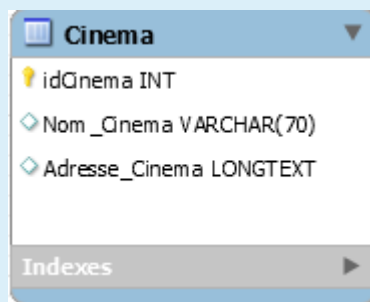
Etablir les relations entre des autres tables :



CREATION ET CONCEPTION DE BASE DE DONNEES AVEC MYSQL WORKBENCH

Définir la structure des tables

Notation simplifiée des tables

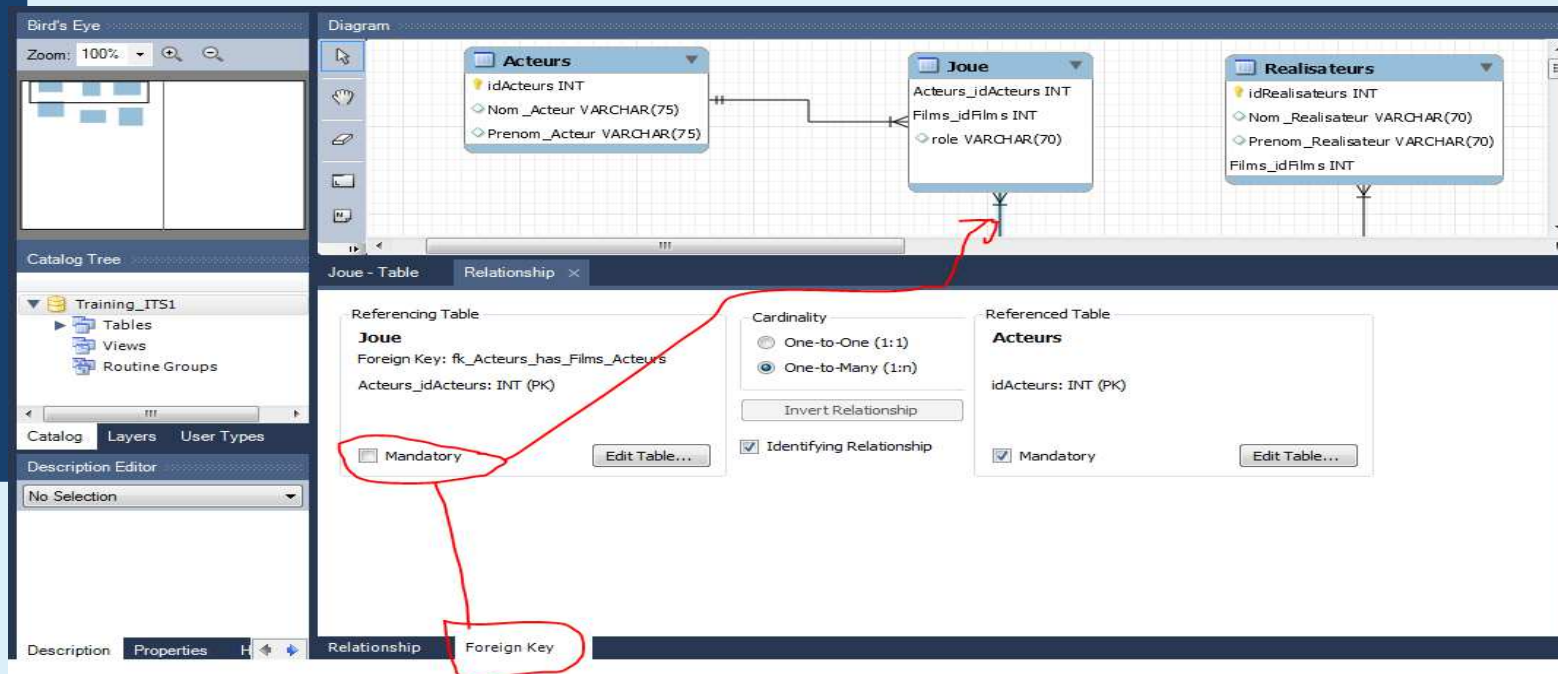


CREATION ET CONCEPTION DE BASE DE DONNEES AVEC MYSQL WORKBENCH

Définir la structure des tables

Cardinalité minimale 0 :

On double-clique sur la patte connectant « Films » et «Joue » puis on décoche la case « Mandatory » associée.



CREATION ET CONCEPTION DE BASE DE DONNEES AVEC MYSQL WORKBENCH

Définir la structure des tables

Actions à effectuer en cas de modification ou de suppression des données d'une table liées:

The screenshot displays the MySQL Workbench interface. The top menu bar includes File, Edit, View, Arrange, Model, Database, Tools, Scripting, and Help. The main workspace shows an EER Diagram with three tables: **Acteurs**, **Joue**, and **Realisateurs**. The **Joue** table is highlighted with a red box. It has columns: **Acteurs_idActeurs** (INT), **Films_idFilms** (INT), and **role** (VARCHAR(70)).

The **Relationship** tab is active, showing the relationship between the **Joue** table and the **Acteurs** table. The **Foreign Key Name** is **fk_Acteurs_has_Films_Acteurs**, and the **Referenced Table** is **Acteurs**. The **Column** is **Acteurs_idActeurs**, and the **Referenced Column** is **idActeurs**. The **Foreign Key Options** are set to **NO ACTION** on update and **CASCADE** on delete. The **Foreign Key Comment** is empty.

The **Modeling Additions** panel on the right shows the following options:

- timestamps**: create_time, update_time
- user**: username, email, password...
- category**: category_id, name

The **Relationship** tab is also highlighted with a red box. The **Table Name** is **Joue**, and the **Schema** is **Training_ITS1**. The **Foreign Key Name** is **fk_Acteurs_has_Films_Acteurs**, and the **Referenced Table** is **Acteurs**. The **Column** is **Acteurs_idActeurs**, and the **Referenced Column** is **idActeurs**. The **Foreign Key Options** are set to **NO ACTION** on update and **CASCADE** on delete. The **Foreign Key Comment** is empty.

The **Foreign Keys** tab is highlighted with a red box. The **Table Name** is **Joue**, and the **Schema** is **Training_ITS1**. The **Foreign Key Name** is **fk_Acteurs_has_Films_Acteurs**, and the **Referenced Table** is **Acteurs**. The **Column** is **Acteurs_idActeurs**, and the **Referenced Column** is **idActeurs**. The **Foreign Key Options** are set to **NO ACTION** on update and **CASCADE** on delete. The **Foreign Key Comment** is empty.

Définir la structure des tables

Actions à effectuer en cas de modification ou de suppression des données d'une table liées:

CASCADE :

- **On Update** : si on modifie une valeur du champ cible de la table liée, tous les enregistrements ayant cette valeur dans le champ de clé étrangère de la table principale sont modifiés en conséquence.
- **On Delete** : si on supprime une valeur du champ cible de la table liée, tous les enregistrements ayant cette valeur dans le champ de clé étrangère de la table principale sont supprimés.

Définir la structure des tables

Actions à effectuer en cas de modification ou de suppression des données d'une table liées:

SET NULL :

- **On Update** : si on modifie une valeur du champ cible de la table liée, tous les enregistrements ayant cette valeur dans le champ de clé étrangère de la table principale sont fixé à **NULL** en conséquence.
- **On Delete** : si on supprime une valeur du champ cible de la table liée, tous les enregistrements ayant cette valeur dans le champ de clé étrangère de la table principale sont fixés à **NULL**.

Définir la structure des tables

Actions à effectuer en cas de modification ou de suppression des données d'une table liées:

NO ACTION :

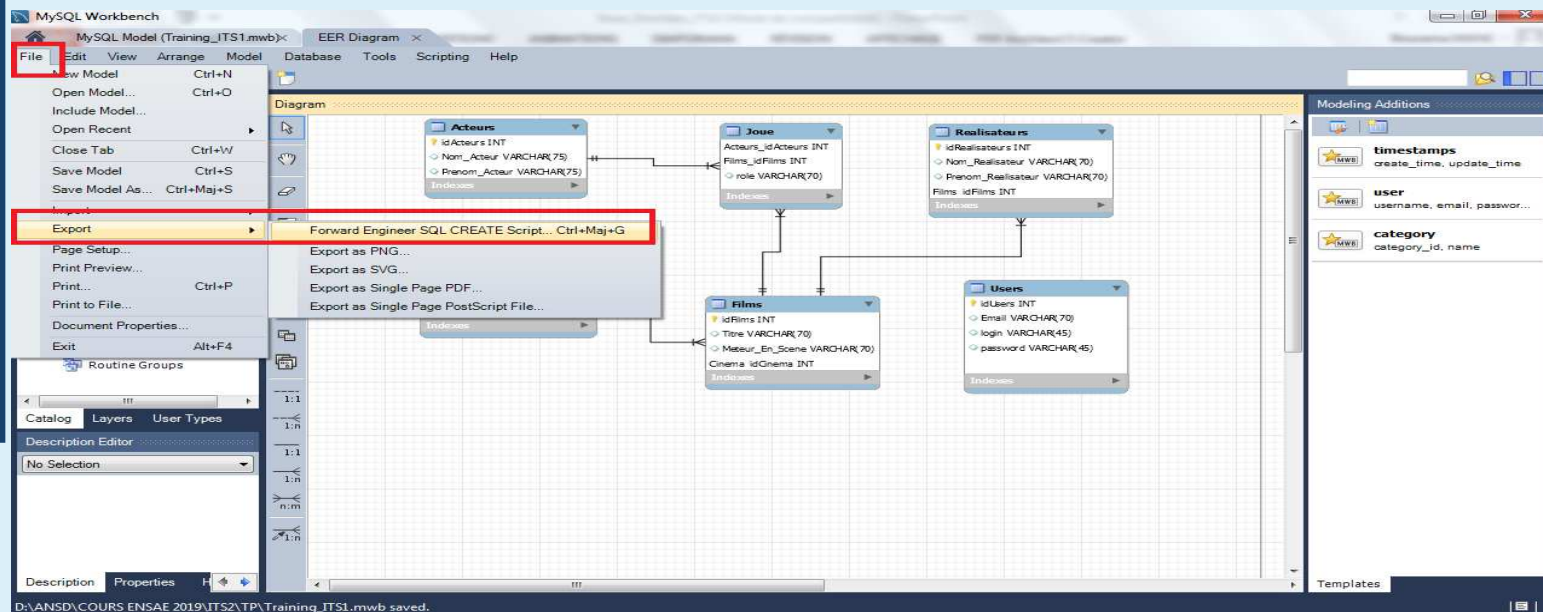
- **On Update** : si on modifie une valeur du champ cible de la table liée, pas de répercussion sur table principale.
- **On Delete** : si on supprime une valeur du champ cible de la table liée, pas de répercussion sur table principale.

CREATION ET CONCEPTION DE BASE DE DONNEES AVEC MYSQL WORKBENCH

Génération du code SQL

Commande « Export »

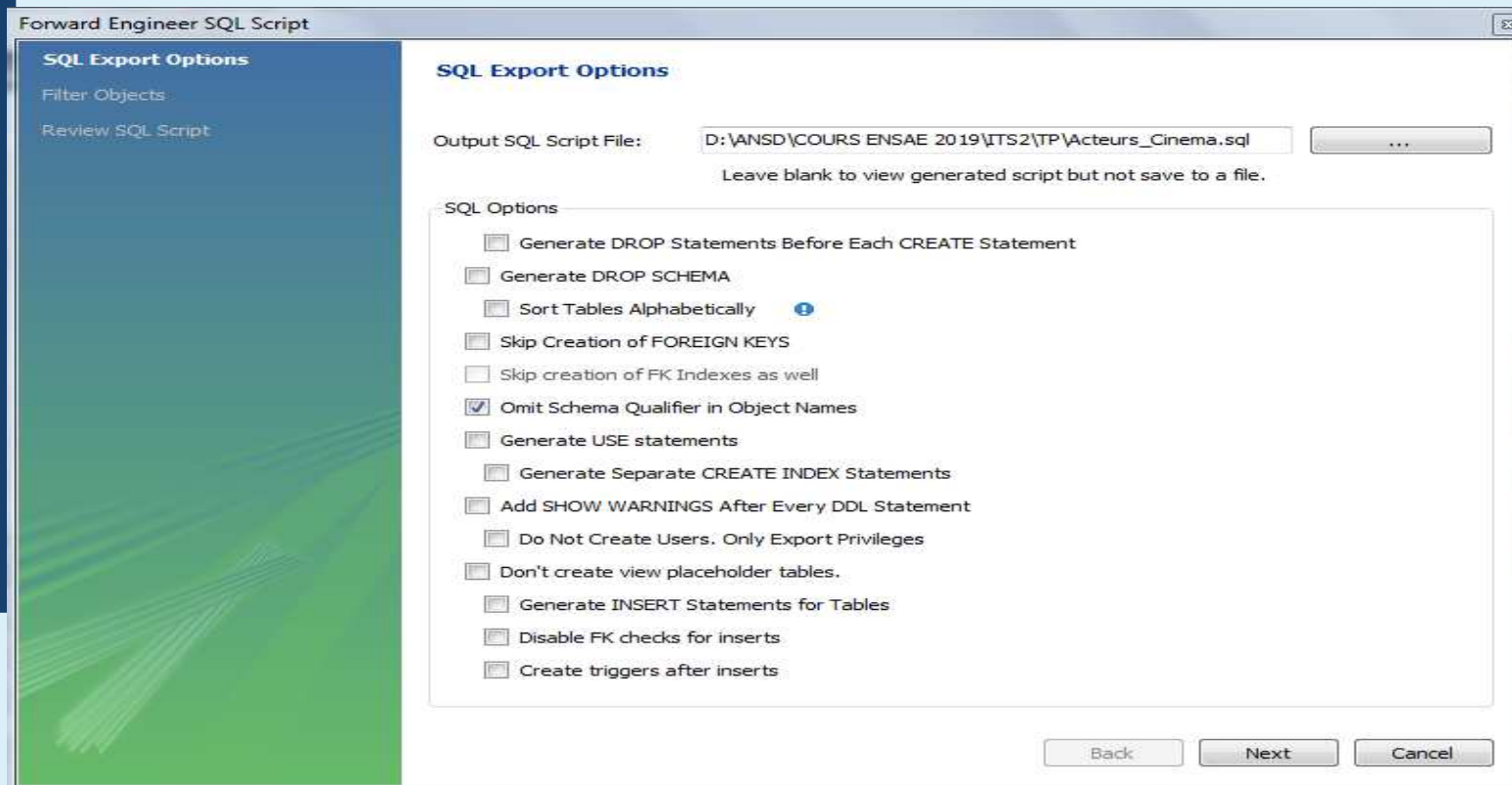
Pour produire le code SQL de création des tables SQL, on utilise la commande « Export » et l'on choisit l'option « Forward Engineer SQL CREATE SCRIPT » :



CREATION ET CONCEPTION DE BASE DE DONNEES AVEC MYSQL WORKBENCH

Génération du code SQL

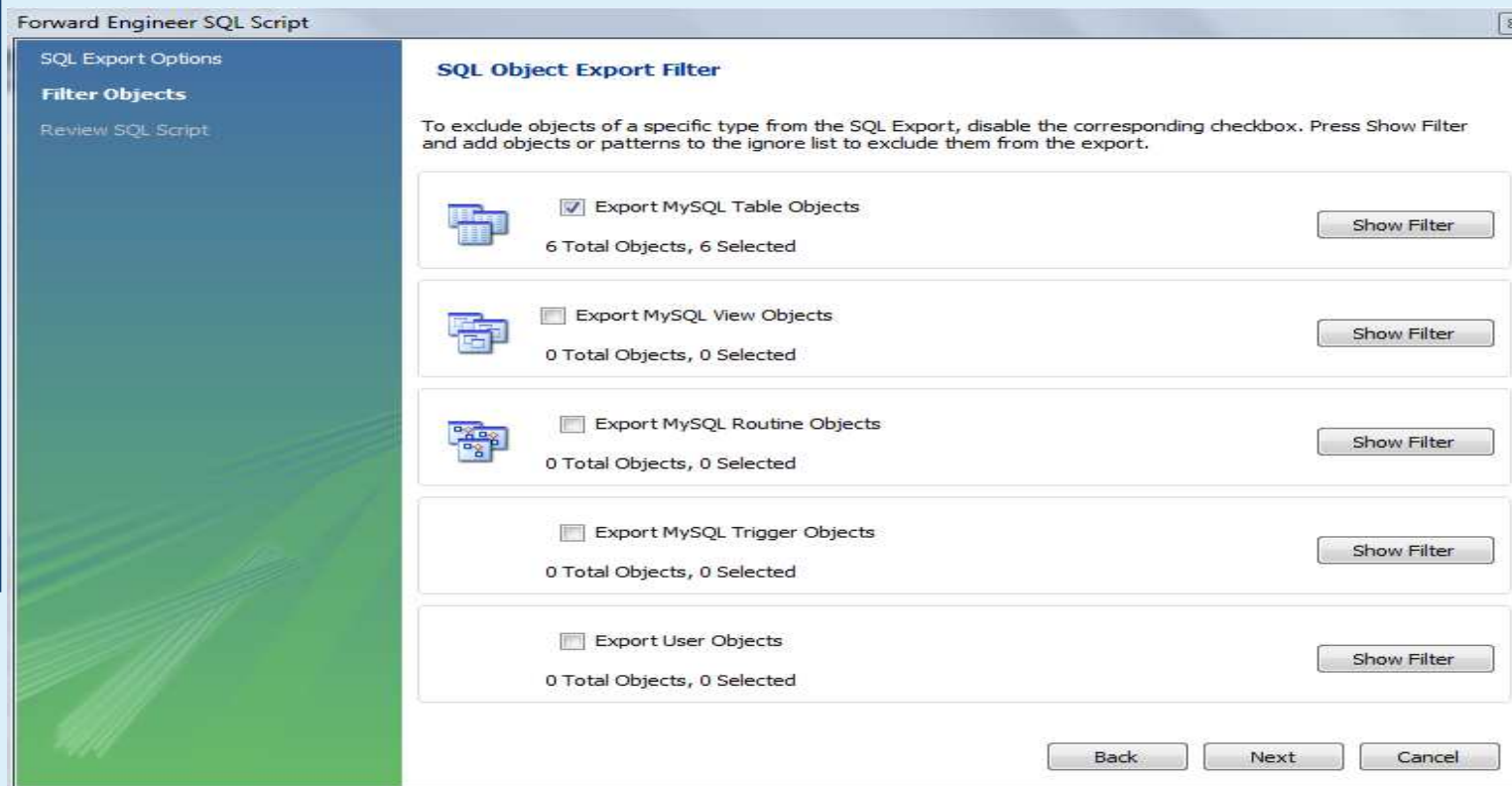
Nom du fichier SQL



CREATION ET CONCEPTION DE BASE DE DONNEES AVEC MYSQL WORKBENCH

Génération du code SQL

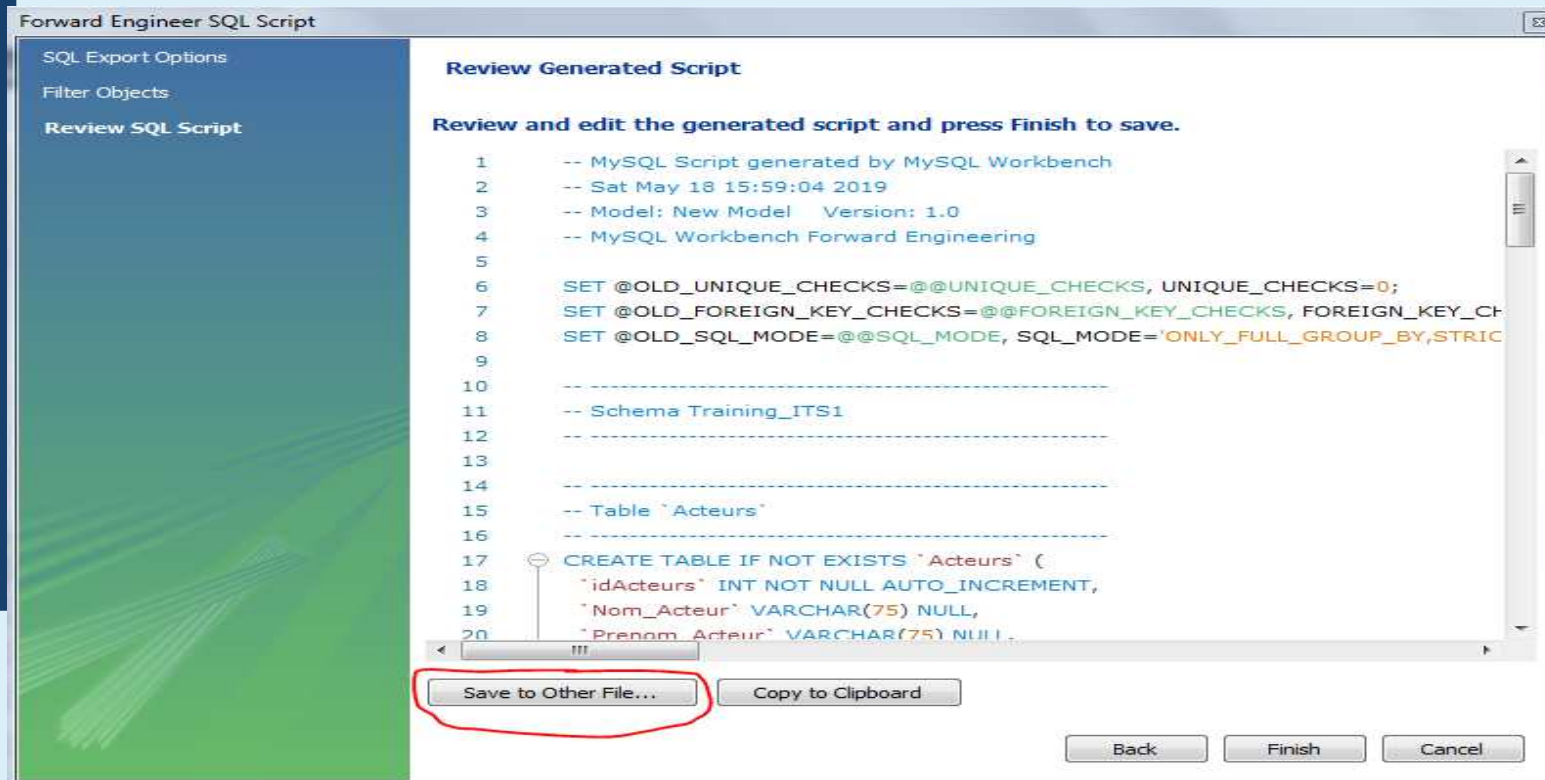
Tables seulement à exporter



CREATION ET CONCEPTION DE BASE DE DONNEES
AVEC MYSQL WORKBENCH

Génération du code SQL

Sauvegarde du fichier SQL





GESTION D'UNE BASE DE DONNEES MYSQL PHPMYADMIN

Création de la base de données par Import du fichier SQL dans PhpMyAdmin

Prérequis :

- Installer WamServer
- Démarrer les service Mysql et Apache
- Situer le fichier SQL à importer
- Lancer PhpMyadmin à l'aide d'un navigateur

Création de la base de données par Import du fichier SQL dans PhpMyAdmin

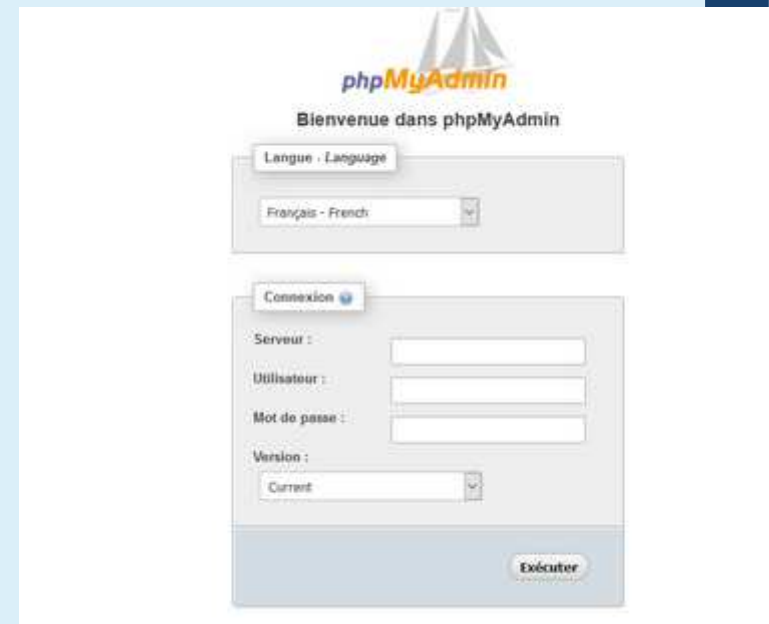
Introduction:

PhpMyadmin est une interface et gestion de base de données MySQL. C'est l'une des interfaces les plus utilisées que ce soit par les hébergeurs gratuits ou payants.

Après installation de WamServer,
PhpMyadmin est accessible à partir

Du lien:

<http://localhost/phpmyadmin>



GESTION D'UNE BASE DE DONNEES MYSQL AVEC PHPMYADMIN

Création de la base de données

localhost/phpmyadmin/

phpMyAdmin

Base de données
(Bases de données) ...

Choisissez une base de données

Serveur: localhost

Bases de données SQL État Variables Jeux de caractères Moteurs Privilèges Processus Exporter

Importer

Bases de données

Base de données	Interclassement
<input type="checkbox"/> information_schema	utf8_general_ci
<input type="checkbox"/> mysql	latin1_swedish_ci
Total: 2	latin1_swedish_ci

Tout cocher / Tout décocher Pour la sélection :

Activer les statistiques

Note: L'activation des statistiques peut causer un trafic important entre le serveur web et le serveur MySQL.

Créer une base de données

Training_ITS2 Interclassement Créer

Ouvrir une nouvelle fenêtre phpMyAdmin

GESTION D'UNE BASE DE DONNEES MYSQL AVEC PHPMYADMIN

Importation des tables

localhost/phpmyadmin/

phpMyAdmin

Base de données
(Bases de données) ...

Training_ITS2 (0)
Aucune table n'a été trouvée dans cette base.

Serveur: localhost Base de données: Training_ITS2

Structure SQL Rechercher Requête Exporter Importer Opérations Privilèges Supprimer

Importer

Fichier à importer

Emplacement du fichier texte **Parcourir...** Acteurs_Cinema.sql (Taille maximum: 2 048Kio)

Jeu de caractères du fichier: utf8

Ces modes de compression seront détectés automatiquement : aucune, gzip, zip

Importation partielle

☒ Permettre l'interruption de l'importation si la limite de temps est sur le point d'être atteinte. Ceci pourrait aider à importer des fichiers volumineux, au détriment du respect des transactions.

Nombre d'enregistrements (requêtes) à ignorer à partir du début 0

Format du fichier d'importation

☒ SQL

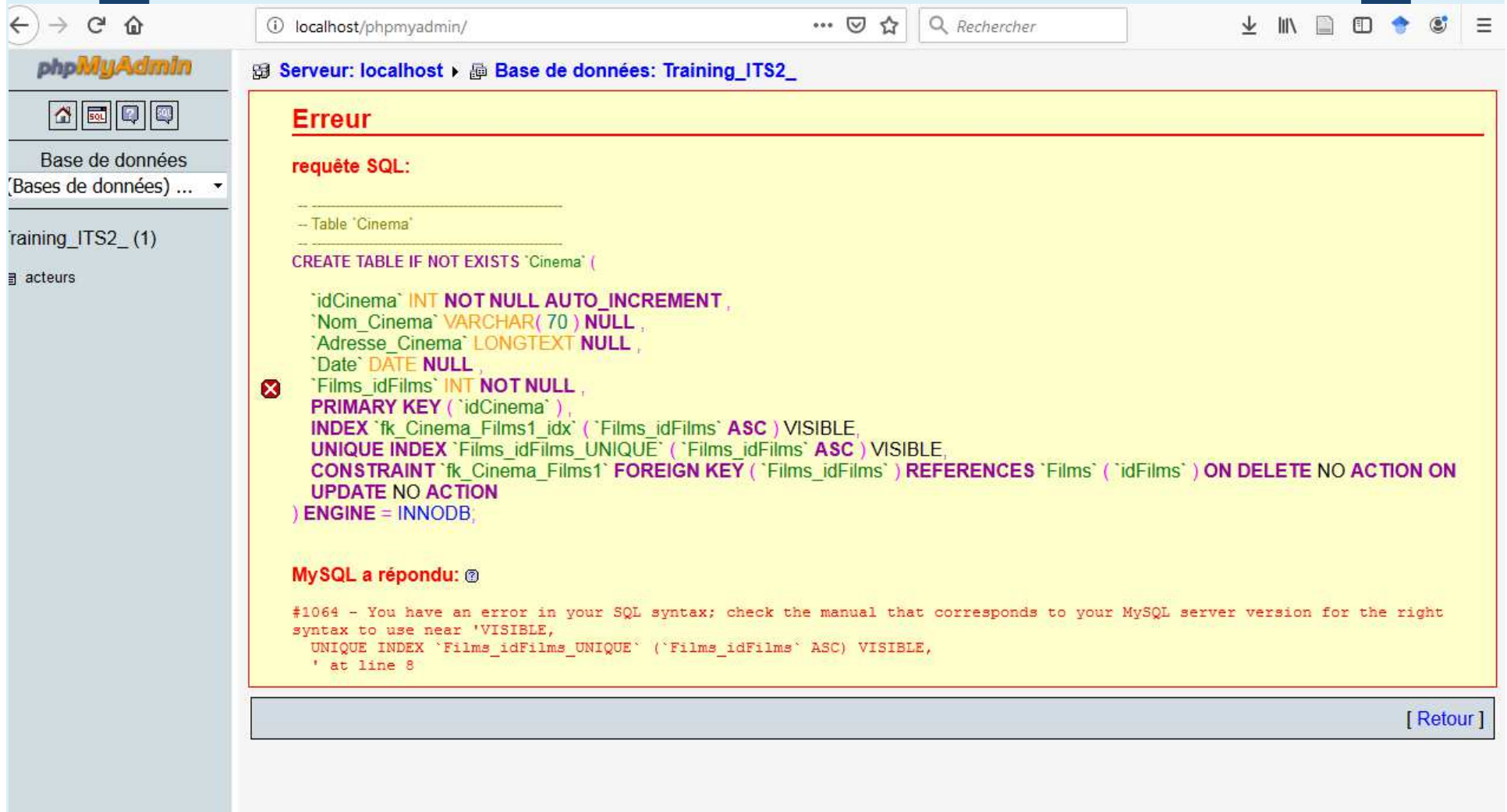
Options SQL

Mode de compatibilité SQL NONE

Exécuter

GESTION D'UNE BASE DE DONNEES MYSQL AVEC PHPMYADMIN

Erreur d'importation à cause de la version de Mysql



The screenshot shows the phpMyAdmin web interface in a browser. The address bar shows 'localhost/phpmyadmin/'. The left sidebar shows the database 'Training_ITS2_' selected. The main area displays an SQL error message. The error message indicates a syntax error in the SQL query, specifically near the 'VISIBLE' keyword in the index definition. The error message is: '#1064 - You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'VISIBLE, UNIQUE INDEX `Films_idFilms_UNIQUE` (`Films_idFilms` ASC) VISIBLE, ' at line 8'. The SQL query being executed is: 'CREATE TABLE IF NOT EXISTS `Cinema` (`idCinema` INT NOT NULL AUTO_INCREMENT, `Nom_Cinema` VARCHAR(70) NULL, `Adresse_Cinema` LONGTEXT NULL, `Date` DATE NULL, `Films_idFilms` INT NOT NULL, PRIMARY KEY (`idCinema`), INDEX `fk_Cinema_Films1_idx` (`Films_idFilms` ASC) VISIBLE, UNIQUE INDEX `Films_idFilms_UNIQUE` (`Films_idFilms` ASC) VISIBLE, CONSTRAINT `fk_Cinema_Films1` FOREIGN KEY (`Films_idFilms`) REFERENCES `Films` (`idFilms`) ON DELETE NO ACTION ON UPDATE NO ACTION) ENGINE = INNODB;'. The error message is displayed in a yellow box with a red border. The SQL query is displayed in a green box with a red border. The error message is displayed in a red box with a red border. The error message is displayed in a red box with a red border.

localhost/phpmyadmin/

phpMyAdmin

Base de données
(Bases de données) ...

Training_ITS2_ (1)

acteurs

Serveur: localhost Base de données: Training_ITS2_

Erreur

requête SQL:

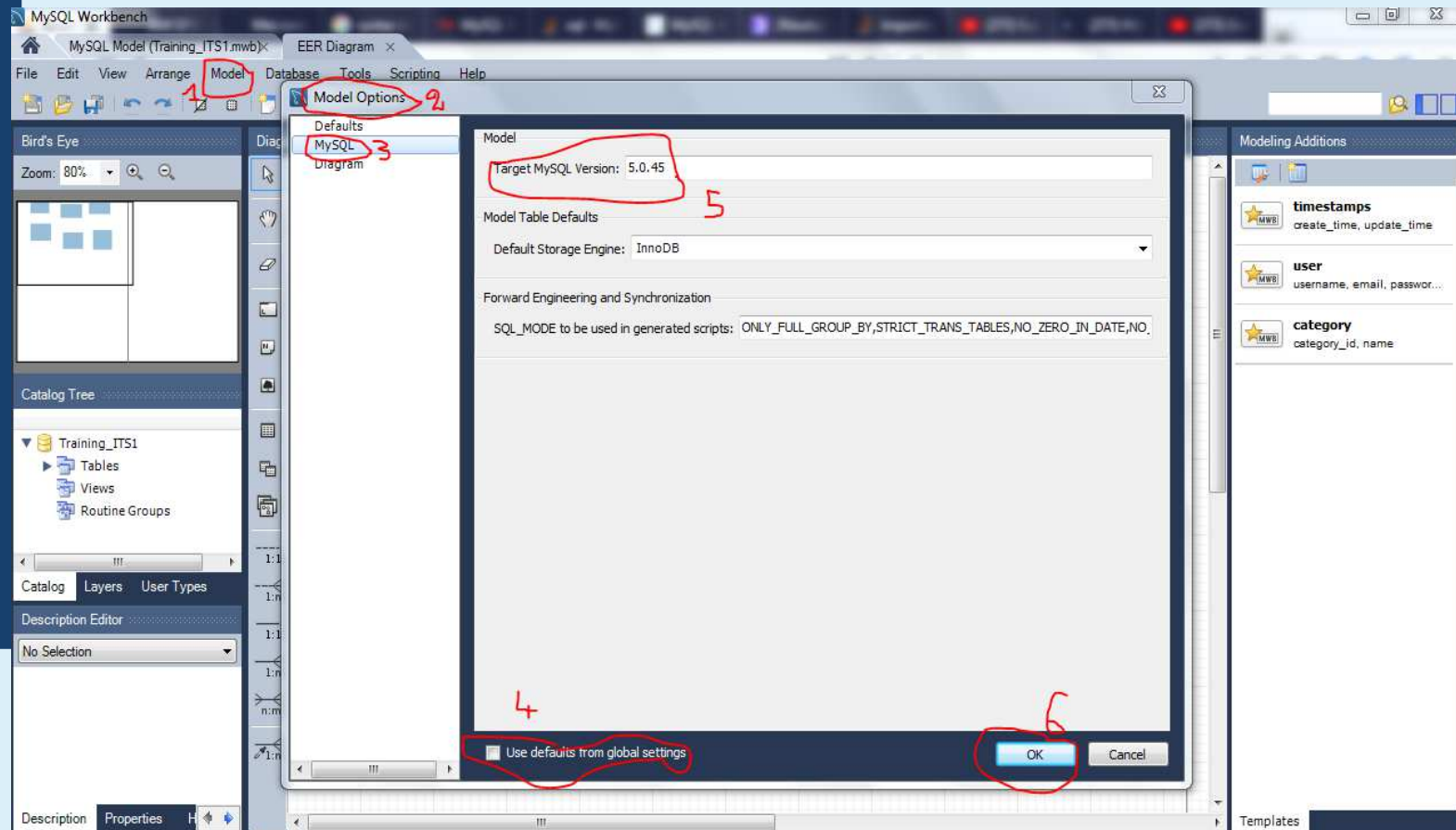
```
-- Table `Cinema`  
CREATE TABLE IF NOT EXISTS `Cinema` (  
  `idCinema` INT NOT NULL AUTO_INCREMENT,  
  `Nom_Cinema` VARCHAR( 70 ) NULL,  
  `Adresse_Cinema` LONGTEXT NULL,  
  `Date` DATE NULL,  
  `Films_idFilms` INT NOT NULL,  
  PRIMARY KEY ( `idCinema` ),  
  INDEX `fk_Cinema_Films1_idx` ( `Films_idFilms` ASC ) VISIBLE,  
  UNIQUE INDEX `Films_idFilms_UNIQUE` ( `Films_idFilms` ASC ) VISIBLE,  
  CONSTRAINT `fk_Cinema_Films1` FOREIGN KEY ( `Films_idFilms` ) REFERENCES `Films` ( `idFilms` ) ON DELETE NO ACTION ON UPDATE NO ACTION  
) ENGINE = INNODB;
```

MySQL a répondu: #1064 - You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'VISIBLE, UNIQUE INDEX `Films_idFilms_UNIQUE` (`Films_idFilms` ASC) VISIBLE, ' at line 8

[Retour]

GESTION D'UNE BASE DE DONNEES MYSQL AVEC PHPMYADMIN

Correction Erreur d'importation à cause de la version de Mysql



GESTION D'UNE BASE DE DONNEES MYSQL AVEC PHPMYADMIN

Importation avec succès

The screenshot shows the phpMyAdmin web interface in a browser. The address bar displays 'localhost/phpmyadmin/'. The page title is 'Serveur: localhost > Base de données: Training_ITS2'. A navigation bar contains links for 'Structure', 'SQL', 'Rechercher', 'Requête', 'Exporter', 'Importer', 'Opérations', 'Privileges', and 'Supprimer'. A yellow message box with an information icon states: 'L'importation s'est terminée avec succès, 12 requêtes exécutées.' This message is circled in red. On the left sidebar, the 'Training_ITS2 (6)' database is selected, and its tables ('acteurs', 'cinema', 'films', 'joue', 'realisateurs', 'users') are listed, also circled in red. Below the message, the 'requête SQL:' section shows the generated SQL script, including 'SET @OLD_UNIQUE_CHECKS = @@UNIQUE_CHECKS'. The 'Importer' section is visible below, with fields for 'Fichier à importer', 'Emplacement du fichier texte', 'Jeu de caractères du fichier', and 'Importation partielle'.

localhost/phpmyadmin/

phpMyAdmin

Base de données
(Bases de données) ...

Training_ITS2 (6)

- acteurs
- cinema
- films
- joue
- realisateurs
- users

Structure SQL Rechercher Requête Exporter Importer Opérations Privileges Supprimer

L'importation s'est terminée avec succès, 12 requêtes exécutées.

requête SQL:

- MySQL Script generated by MySQL Workbench
- Tue May 28 10:17:27 2019
- Model: New Model Version: 1.0
- MySQL Workbench Forward Engineering
- SET @OLD_UNIQUE_CHECKS = @@UNIQUE_CHECKS

Importer

Fichier à importer

Emplacement du fichier texte Parcourir... Aucun fichier sélectionné. (Taille maximum: 2 048Kio)

Jeu de caractères du fichier: utf8

Ces modes de compression seront détectés automatiquement : aucune, gzip, zip

Importation partielle

☒ Permettre l'interruption de l'importation si la limite de temps est sur le point d'être atteinte. Ceci pourrait aider à importer des fichiers volumineux, au détriment du respect des transactions.

Nombre d'enregistrements (requêtes) à ignorer à partir du début 0

Format du fichier d'importation

SQL

GESTION D'UNE BASE DE DONNEES MYSQL AVEC PHPMYADMIN

Insertion de données dans une table

localhost/phpmyadmin/

Rechercher

Serveur: localhost Base de données: training_its2 Table: realisateurs "InnoDB free: 4096 kB"

Afficher Structure SQL Rechercher **insérer** Exporter Importer Opérations Vider Supprimer

Champ	Type	Fonction	Null	Valeur
idRealisateurs	int(11)			
Nom_Realisateur	varchar(70)		<input checked="" type="checkbox"/>	
Prenom_Realisateur	varchar(70)		<input checked="" type="checkbox"/>	

☒ Ignorer

Champ	Type	Fonction	Null	Valeur
idRealisateurs	int(11)			
Nom_Realisateur	varchar(70)		<input checked="" type="checkbox"/>	
Prenom_Realisateur	varchar(70)		<input checked="" type="checkbox"/>	

Sauvegarder un nouvel enregistrement et ensuite Retourner à la page précédente

Exécuter Réinitialiser les valeurs

Attention : idRealisateurs est en Auto_Incrément

Autre méthode d'insertion de données dans une table

L'ajout (ou insertion) d'une donnée dans une table peut aussi se faire via l'instruction SQL *INSERT INTO* suivi du nom de la table et des valeurs à insérer.

Syntaxe:

```
INSERT INTO mabase.matable (champ1, champ2, champ3) VALUES  
(chiffre1, chiffre2, 'chaîne de caractère')
```

Exemple:

```
INSERT INTO `training_its2`.`realisateurs` (`idRealisateurs` ,  
`Nom_Realisateur` , `Prenom_Realisateur` ) VALUES (NULL , 'CAMARA',  
'Demba ');
```

Attention : idRealisateurs est en Auto_Incrément

GESTION D'UNE BASE DE DONNEES MYSQL AVEC PHPMYADMIN

Autre méthode d'insertion de données dans une table

The screenshot shows the phpMyAdmin interface for a MySQL database named 'training_its2'. The 'realisateurs' table is selected. The SQL tab is active, and a query is entered in the text area. The query is: `INSERT INTO `training_its2`.`realisateurs` (`idRealisateurs`, `Nom_Realisateur`, `Prenom_Realisateur`) VALUES (NULL, 'MBODJ', 'Alexandre');`. The query is circled in red with the number '3' next to it. The 'Exécuter' button is circled in red with the number '4' next to it. The 'realisateurs' table is circled in red with the number '1' next to it. The 'SQL' tab is circled in red with the number '2' next to it.

localhost/phpmyadmin/

Rechercher

Serveur: localhost Base de données: training_its2 Table: realisateurs "InnoDB free: 4096 kB"

Afficher Structure SQL Rechercher Insérer Exporter Importer Opérations Vider Supprimer

Base de données
training_its2 (6)

training_its2 (6)

- acteurs
- cinema
- films
- joue
- realisateurs**
- users

Exécuter une ou des requêtes sur la base training_its2:

```
INSERT INTO `training_its2`.`realisateurs` (`idRealisateurs`, `Nom_Realisateur`, `Prenom_Realisateur`) VALUES (NULL, 'MBODJ', 'Alexandre');
```

Champs
idRealisateurs
Nom_Realisateur
Prenom_Realisateur

[Délimiteur ;] ☒ Réafficher la requête après exécution

Exécuter

Afficher les données de la table

The screenshot shows the phpMyAdmin interface with the 'Afficher' tab selected. The left sidebar shows the 'realisateurs' database selected. The main area displays the SQL query 'SELECT * FROM `realisateurs` LIMIT 0, 30' and the resulting table of 4 records. The table has columns 'idRealisateurs', 'Nom_Realisateur', and 'Prenom_Realisateur'. The records are: 1 CAMARA Demba, 2 DEME Saliou, 3 BADJI Aliou, and 4 MBODJ Alexandre. The interface is circled in red.

**GESTION D'UNE BASE DE DONNEES
MYSQL AVEC PHPMYADMIN**

Insertion des données des autres tables

TP