

## A. Problèmes avec l'utilisation d'une table unique.

---

- Redondances des données
  - Efficacité des requêtes
  - Maintenance et la mise à jour.
- 

## B. Les Tables relationnelles

---

### Principe

- Il s'avère plus efficace de ranger les données dans plusieurs tables et les relier entre elles.
- Dans l'exemple ci-dessous le schéma illustre le principe.
  - Pour chaque enregistrement de la table **Film**, on a un champ réservé à un index. Appelons le **genre\_id**.
  - Chaque index doit correspondre à la clé primaire d'un enregistrement de la table **Genre**.
  - **NB** : (Pour ce type de relation, on verra qu'il faut faire de cet index une clé **clé étrangère**).

Film		
id	titre	genre_id
1	Leon	2
2	Mama mia	3
3	Les Tuche	3

Genre	
id	nom
1	policier
2	action
3	comédie

### Termes de relations :

- Un film ou plusieurs films appartiennent à un genre. On appelle cette relation **Many to One**
  - Un genre peut se trouver dans un ou plusieurs films. On appelle cette relation **One to Many**
- 

## C. Tables relationnelles de la base db\_films

---


- Définissons nos tables. **nb**: Majuscule, pas d'accents ni d'espaces.
  - La table **Genre** avec les champs : (id, nom)
  - La table **Pays** avec les champs (id, nom)
  - la table **Realisateur** avec les champs (id, nom)
  - la table **Film** avec les champs (id, titre, sorti, id\_realisateur, id\_pays, id\_genre)

### Création des tables

- On reprend notre base de données en commençant par les tables sans clé étrangère.
- Il n'est pas obligatoire de les alimenter à ce stade, mais on peut aussi le faire.

## La table Genre


Ici, la structure de la table Genre, pour l'exemple.

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
<input type="checkbox"/>	1	id 	int(2)			No	None		AUTO_INCREMENT
<input type="checkbox"/>	2	nom	varchar(100)	utf8_general_ci		No	None		

**NB** Faites de même pour les autres tables

## La table Film

En dernier, on crée la table film avec les champs qui vont pointer sur les clés primaires des autres tables (*clés étrangères*).

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
<input type="checkbox"/>	1	id 	int(11)			No	None		AUTO_INCREMENT
<input type="checkbox"/>	2	titre	varchar(255)	utf8_general_ci		No	None		
<input type="checkbox"/>	3	sorti	date			No	None		
<input type="checkbox"/>	4	genre_id	int(2)			No	None		
<input type="checkbox"/>	5	pays_id	int(2)			No	None		
<input type="checkbox"/>	6	realisateur_id	int(2)			No	None		

## Test d'insertion dans Film

Quand on fait un test d'insertion avec l'interface PHPMyAdmin, on voit rien de particulier qui nous montre quelque relation avec d'autres tables.

**Normal !** Nous n'avons pas encore relié les champs `genre_id`, `realisateur_id`, `pays_id` avec les tables respectives.

## Vues relationnelles

Depuis l'onglet structure, nous voyons un bouton **relation view**, il ouvre une page formulaire laquelle nous permet de relier nos 2 tables.

Table structure

Relation view

Foreign key constraints

Actions

Constraint properties

Column

Foreign key constraint (INNODB)

Drop

FK\_film\_pays

ON DELETE

NO ACTION

ON UPDATE

NO ACTION

pays\_id

db\_movies

pays

id

+ Add column

On constate maintenant sur la page de la structure que notre champ **pays\_id** est devenu un index. Regardez aussi la ligne plus bas, section : **indexes** !

Table structure
Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
<input type="checkbox"/> 1	<b>id</b>	int(11)			No	None		AUTO_INCREMENT
<input type="checkbox"/> 2	<b>titre</b>	varchar(255)	utf8_general_ci		No	None		
<input type="checkbox"/> 3	<b>sorti</b>	date			No	None		
<input type="checkbox"/> 4	<b>genre_id</b>	int(2)			No	None		
<input type="checkbox"/> 5	<b>pays_id</b>	int(2)			No	None		
<input type="checkbox"/> 6	<b>realisateur_id</b>	int(2)			No	None		

☐ Check all
 With selected:
 Browse
Change
Drop
Primary
Unique

Print
Propose table structure
Move columns
Normalize

Add

column(s)
 
Go

**Indexes**

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit  Drop	<b>PRIMARY</b>	BTREE	Yes	No	id	0	A	No	
Edit  Drop	<b>fk_film_pays</b>	BTREE	No	No	pays_id	0	A	No	

Faites la même chose pour les autres champs index. On doit avoir comme résultat :

Table structure
Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	<b>id</b>	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/> 2	<b>titre</b>	varchar(255)	utf8_general_ci		No	None			Change  Drop  More
<input type="checkbox"/> 3	<b>sorti</b>	date			No	None			Change  Drop  More
<input type="checkbox"/> 4	<b>genre_id</b>	int(2)			No	None			Change  Drop  More
<input type="checkbox"/> 5	<b>pays_id</b>	int(2)			No	None			Change  Drop  More
<input type="checkbox"/> 6	<b>realisateur_id</b>	int(2)			No	None			Change  Drop  More

☐ Check all
 With selected:
 Browse
Change
Drop
Primary
Unique
Index
Fulltext

Print
Propose table structure
Move columns
Normalize

Add

column(s)
 
Go

**Indexes**

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit  Drop	<b>PRIMARY</b>	BTREE	Yes	No	id	0	A	No	
Edit  Drop	<b>fk_film_pays</b>	BTREE	No	No	pays_id	0	A	No	
Edit  Drop	<b>fk_film_realisateur</b>	BTREE	No	No	realisateur_id	0	A	No	
Edit  Drop	<b>fk_film_genre</b>	BTREE	No	No	genre_id	0	A	No	

## Insertion dans Pays

On insère deux pays pour vérifier la relation

+ Options

				id	nom
<input type="checkbox"/>	Edit	Copy	Delete	1	France
<input type="checkbox"/>	Edit	Copy	Delete	2	Espagne

## Page d'insertion de Film maintenant.

Ouvrez le sélecteur du champ `pays_id` et constatez !

On peut maintenant relier un pays à un film.

`pays_id`    `int(2)`       

## Insérer les données

Commencez par `Pays`, `Genre` et `Realisateur`, puis continuer avec `Film`

```
-- GENRE
INSERT INTO
  genre (id,nom)
VALUES
  (null,'drame'), (null,'comédie'), (null,'western'), (null,'science fiction'),
  (null,'aventure'), (null,'animation')

-- PAYS
INSERT INTO
  pays (id,nom)
VALUES
  (null,'Japon'), (null,'France'), (null,'Espagne'), (null,'Etats Unis')

-- REALISATEUR
INSERT INTO
  realisateur (id, nom )
VALUES
  (null, 'Emmanuelle Bercot'), (null,'Cédric Klapisch'), (null,'Quentin
Tarantino'),
  (null,'Pedro Almodovar'),(null,'Georges Lucas'), (null,'Hayao Miyazaki');

-- FILM
-- A ce stade, il est plus simple de passer par l'interface.
```

## D. Sélections multi tables

Pour sélectionner de données qui se trouvent sur plusieurs tables, nous devons utiliser des requêtes avec de jointures qui font correspondre les clés étrangères avec des clé primaires.

**NB** IL existe plusieurs façons. Voyons cela dans une autre base de données.

## Une base de test avec Deux tables

Créez une base de données `db_membres` avec la relation entre les deux

```
Table Membre (id, prenom, code_id)
Table Affectation (id, code)
```

Ce sera aussi l'occasion de voir une autre technique d'importation de données.

### Format CSV

IL est possible de créer le contenu à partir d'un tableur du type Excel.

Importer depuis un fichier csv

`affectation.csv`

	A	B	C	D
1	1	1000		
2	2	1001		
3	3	1002		
4	4	1003		
5				

`membre.csv`

	A	B	C	D	E
1	1	louis	4		
2	2	Anna	1		
3	3	Paul	2		
4	4	Marie	1		
5					
6					

## Jointures et Différences de résultat

L'objectif est de récupérer les membres dont le code n'est pas nul.

left join

```
-- toutes les lignes de la table membre sont retournées, même s'il n'y a pas de
correspondance dans la table affectation
SELECT
```

```
membres.prenom, affectation.code from membres
left join affectation on membres.code_id = affectation.id and affectation.code
> 0
```

## inner join

```
-- toutes les lignes de la table membres sont retournées, même s'il n'y a pas de
correspondante dans la table affectation
SELECT
    membres.prenom, affectation.code from membres
    inner join affectation on membres.code_id = affectation.id and
affectation.code > 0
```

## Sélection dans la base des films

### Selection avec jointure de 2 tables

```
SELECT F.titre, F.sorti, P.nom from film F
inner join pays P ON F.pays_id = P.id and P.nom = 'france'
```

### Selection avec jointure 3 tables

```
SELECT
    F.titre,
    F.sorti,
    P.nom,
    R.nom
from film F
inner join pays P ON F.pays_id = P.id
inner join realisateur R ON F.realisateur_id = R.id
and P.nom = 'france'
```

## E. Exercice de groupe en classe

---

### Sujet

- Vous définissez la base de données du thème de votre choix
- Exemples de thèmes : *Musées, Capitales, Romans, Joueurs, albums, peintures, ...*

### Contraintes SQL

- Au moins deux tables reliées par des clé étrangères

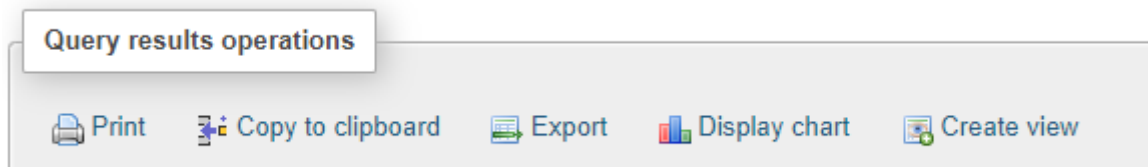
## Type de rendu

- ici, pas de fichiers sql, mais un document pdf combinant une présentation et différentes phases de construction, relation et requêtes

### Comment faire ?

- Depuis PhpMyAdmin, il est possible d'imprimer en pdf structures, requêtes et résultats.
- Exemple avec la table film. En dessous la liste des film, on trouve une série de liens,

### Premier print



### deuxième print

- Pour enregistrer au format pdf

✓ Showing rows 0 - 7 (8 total, Query took 0.0013 seconds.)

```
SELECT * FROM `film`
```

id	titre	sorti	genre_id	pays_id	realisateur_id
1	La tête haute	2015-10-21	3	1	1
2	Ce qui nous lie	2017-06-04	4	1	2
3	Once upon a time	2019-06-26	4	6	3
4	Django Unchained	2013-01-13	5	6	3
5	Tout sur ma mère	1999-05-19	3	5	4
6	Star Wars : Le Réveil de la Force	2015-12-16	6	6	5
7	Les aventuriers de l'arche perdue	1981-09-16	7	6	5
8	Le château ambulant	2005-01-12	8	3	6

Back Print

### Exemple de SELECT

Run SQL query/queries on table db\_movies.film: ?

```
1 SELECT F.titre,F.sorti, P.nom from film F
2 inner join pays P ON F.pays_id = P.id and P.nom = 'france'
```

SELECT \*

SELECT

INSERT

UPDATE

DELETE

Clear

Format

☐ Bind parameters ?


[ Delimiter ; ]


☐ Show this query here again

☐ Retain query box

☐ Rollback when

Hide query box

 Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete

 Showing rows 0 - 1 (2 total, Query took 0.0056 seconds.)

```
SELECT F.titre,F.sorti, P.nom from film F inner join pays P ON F.pays_id = P.id and P.nom = 'france'
```

titre	sorti	nom
La tête haute	2015-10-21	France
Ce qui nous lie	2017-06-04	France

Back

Print

Cliquez sur le

bouton print pour enregistrer le tout en pdf

8 / 8