

# CARDINALITES

Comprendre les relations et les  
cardinalités



# Exercices #2

Dans ces exercices, il conviendra de savoir lire des demandes **moins techniques** et donc **plus proche de celle du client** pour en déduire les cardinalités entre les différentes entités/tables.

Un utilisateur commande des produits sur un site e-commerce. On veut stocker quel produit est dans quelle commande. Comment faire ? (15 min)



Comment stocker la relation  
« composer » entre « Produit »  
et « Commande » ?

Par exemple, une bonne réponse peut être :

- Un « Produit » peut « composer » combien de « Commande » ? Plusieurs commandes
- Une « Commande » peut être « composée » de combien de « Produit » ? Plusieurs produits

C'est donc une relation ManyToMany (ou n-n) :

- Plusieurs commandes
- Plusieurs produits

Pour une relation ManyToMany, nous allons créer une nouvelle structure.



# On stocke la relation « composer »

id	...
ref	...

commande

commande_id	...
produit_id	...

composer\_commande\_produit

id	...
titre	...
prix	...

produit

On réalise un site comme leboncoin.  
On veut stocker quel particulier a  
posté quelle annonce. Comment  
faire ? (15 min)



Comment stocker la relation  
« poster » entre « Particulier » et  
« Annonce » ?

Par exemple, une bonne réponse peut être :

- Un « Particulier » peut « poster » combien de « Annonce » ? Plusieurs annonces
- Une « Annonce » peut être « posté » par combien de « Particulier » ? Un seul particulier

C'est donc une relation OneToMany (ou 1-n) :

- Plusieurs annonces
- Un seul particulier

Pour stocker la relation 1 particulier – n annonces, on va créer une nouvelle colonne à la patte n, donc message :

id	...
nom	...
email	...

particulier

id	...
objet	...
contenu	...
post_particulier_id	...

annonce

On réalise un site pour le  
gouvernement.

On veut stocker quelle voiture a été  
immatriculée à quelle préfecture.

Comment faire ? (15 min)



Comment stocker la relation  
« immatriculer » entre  
« Voiture » et « Préfecture » ?



Par exemple, une bonne réponse peut être :

- Une « Préfecture » peut « immatriculer » combien de « Voiture » ? Plusieurs voitures
- Une « Voiture » peut être «immatriculée » par combien de « Préfecture » ? Plusieurs préfectures (en cas de changement de propriétaire)

C'est donc une relation ManyToMany (ou n-n) :

- Plusieurs voitures
- Plusieurs préfectures

Pour une relation ManyToMany, nous allons créer une nouvelle structure.

# On stocke la relation « immatriculer »

id	...
ville	...

prefecture

prefecture_id	...
voiture_id	...

immatriculation\_voiture\_prefecture

id	...
modele	...
puissance	...

voiture

On réalise un site comme  
booking.com.

On veut stocker quel particulier a  
réservé quelle chambre. Comment  
faire ? (15 min)



Comment stocker la relation  
« réserver » entre « Particulier »  
et « Chambre » ?

Par exemple, une bonne réponse peut être :

- Un « Particulier » peut « réserver » combien de « Chambre » ? Plusieurs chambres
- Une « Chambre » peut être « réservée » par combien de « Particulier » ? Plusieurs particuliers (1 particulier à une date, un autre le lendemain par exemple)

C'est donc une relation ManyToMany (ou n-n) :

- Plusieurs chambres
- Plusieurs particuliers

Pour une relation ManyToMany, nous allons créer une nouvelle structure.

# On stocke la relation « reserver »

id	...
numero	...

chambre

chambre_id	...
particulier_id	...

reserver\_particulier\_chambre

id	...
email	...
nom	...

particulier



