

Les types de relations en base de données

Introduction

Les relations constituent la principale caractéristique des bases de données relationnelles. Les différents types de relations contrôlent la manière dont les données et les tables sont liées les unes aux autres. L'établissement de liens entre les tables par le biais de relations rend ce type de base de données unique, et la connaissance du fonctionnement des relations permet d'élargir les capacités de modélisation des bases de données.

Cet article explique les différents types de relations et leur fonctionnement à l'aide d'exemples pratiques.

Comment les relations s'établissent-elles ?

Les deux éléments suivants définissent la manière dont une relation de base de données est établie :

- **Clé primaire** : Colonne dont la valeur identifie de manière unique un enregistrement de la table.
- **Clé étrangère** : Colonne dont les valeurs font référence à la clé primaire d'une autre table.

Une connexion entre une table parent et une table enfant existe en référençant la clé primaire de la table parent.

Le comportement de la référence définit le type de relation entre deux tables de la base de données.

Une base de données utilise les relations entre deux tables par le biais d'instructions JOIN dans un langage de requête structuré (SQL).

Pourquoi les relations sont-elles importantes dans une base de données ?

Les relations dans une base de données permettent de créer des informations significatives. Par conséquent, les relations dans une base de données se traduisent par :

- **Une réduction de la redondance des données** : Les relations permettent de référencer les informations stockées dans les tables existantes, réduisant ainsi les répétitions.
- **Des bases de données mieux organisées** : Les relations permettent de mettre en œuvre des techniques de normalisation des bases de données. La normalisation permet d'obtenir une base de données mieux organisée et plus robuste.
- **Intégrité référentielle** : Au fur et à mesure que les bases de données se développent, les jointures, les requêtes et les tris deviennent coûteux. Les relations permettent de réduire le nombre de transactions et d'améliorer la validité des données.

Les relations établies entre les bases de données garantissent la viabilité, la flexibilité et la stabilité d'un SGBDR (Système de gestion de base de données relationnelles).

Les types de relations dans les bases de données

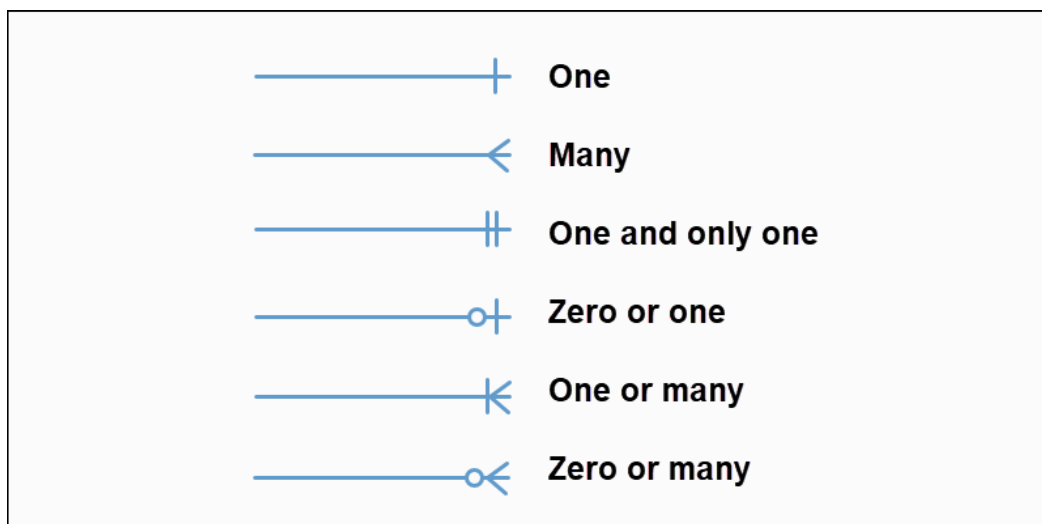
Une base de données relationnelle met en œuvre trois types de relations différentes :

1. Une à une (1:1)
2. Une à plusieurs (1:N)
3. Plusieurs à plusieurs (N:N)

Une ligne reliant deux tables représente une relation, tandis que les symboles à l'extrémité de la ligne représentent le type de relation exact.

Par exemple, dans les diagrammes ER (Diagramme de relations entre entités), les cardinalités des relations "une" et "plusieurs" apparaissent sous la forme des symboles suivants :

- Une ligne perpendiculaire (un).
- Patte d'oie (plusieurs).
- Un cercle (zéro).



Les symboles se combinent pour former des types de relations complexes

1. Relation de type "un à un" dans une base de données

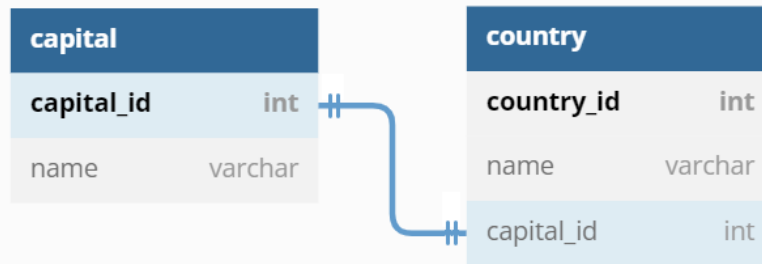
Une relation "un à un" (1:1) dans une base de données comporte un enregistrement de chaque côté de la relation. Chaque clé primaire se rapporte à au plus une entrée d'une autre table, ce qui rend la clé étrangère unique.

Exemple de relation "un à un"

Un exemple simple de relation "un à un" est celui des capitales. Un pays (ou un État) n'a qu'une seule capitale, et une capitale n'appartient qu'à un seul pays (ou État).

Deux tables contenant des informations sur les pays et les capitales sont reliées dans une base de données à l'aide d'une clé primaire. Par exemple, lorsqu'il est ajouté à la table des pays, l'identifiant unique d'une capitale (sa clé primaire) devient une clé étrangère, créant ainsi une relation.

One to one database relationship



Dans ce cas, la relation "un à un" est obligatoire. Chaque pays doit avoir une capitale unique et la clé étrangère doit être unique pour garantir la relation 1:1.

Note : D'autres façons de créer une relation 1:1 consisteraient à utiliser le `country_id` comme identifiant unique pour la table `capital`, ou à référencer le `country_id` comme clé étrangère unique.

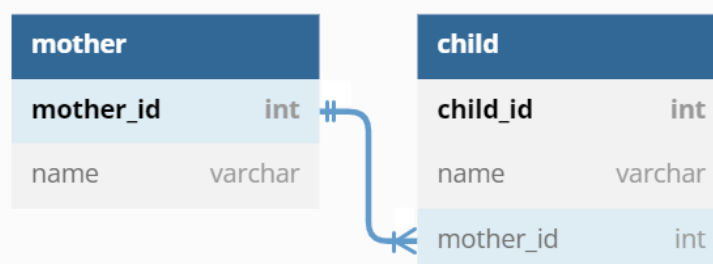
2. Relation de type "un à plusieurs" dans une base de données

Dans une base de données, une relation de type "un à plusieurs" (1:N) comporte une seule entrée d'un côté et plusieurs entrées de l'autre. Chaque clé primaire correspond à un ou plusieurs enregistrements d'une autre table. Dans ce cas, la clé étrangère n'est pas unique. Les relations "un à plusieurs" sont naturelles et apparaissent souvent comme une connexion logique dans la modélisation des bases de données.

Exemple de relation de type "un à plusieurs" :

Le lien entre une mère et ses enfants est un exemple de relation de type "un à plusieurs". Une mère peut avoir de nombreux enfants, mais chaque enfant n'appartient qu'à une seule mère. Une base de données contenant deux tables avec des informations sur les mères et les enfants se connecte à l'aide d'une clé primaire. Lorsqu'il est ajouté à la table des enfants, l'identifiant unique d'une mère devient une clé étrangère. Différents enfants peuvent avoir la même mère.

One to many database relationship



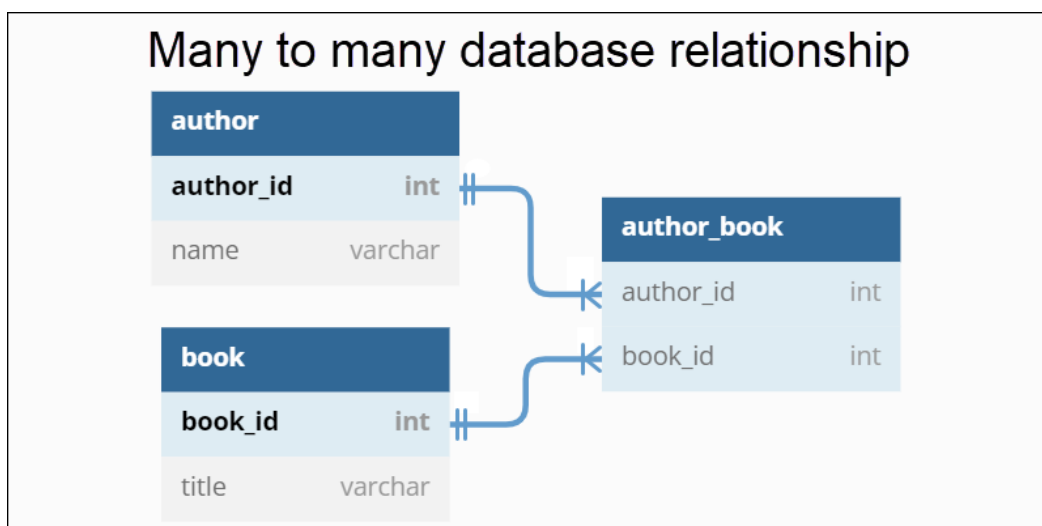
La relation est obligatoire aux deux extrémités, et au moins une entrée doit exister dans les deux tables pour établir une connexion.

3. Relation de type "plusieurs à plusieurs" dans une base de données

Les relations de plusieurs à plusieurs (N:N) ou (M:M) dans une base de données comportent plusieurs entrées aux deux extrémités de la relation. Étant donné que de nombreuses entrées peuvent exister aux deux extrémités, une solution standard consiste à créer une table d'association (jonction, jointure) avec des clés étrangères provenant des deux tables. Les relations de plusieurs à plusieurs sont une pratique courante dans les bases de données relationnelles mises en œuvre dans les technologies web, telles que les sites de commerce électronique.

Exemple de relation de plusieurs à plusieurs :

Il existe une relation de plusieurs à plusieurs entre les livres et les auteurs. Par exemple, un seul livre peut avoir plusieurs auteurs. De même, un auteur unique peut avoir plusieurs livres. S'il existe une table contenant des livres et une autre contenant des auteurs, la meilleure façon d'établir la relation entre les deux est de créer une nouvelle table. La nouvelle table possède des clés étrangères provenant des deux tables mères, ce qui crée une relation de plusieurs à plusieurs.



L'exécution de différents types de requêtes JOIN permet de récupérer efficacement les données des deux tables tout en protégeant les tables d'origine contre les redondances.

Conclusion

Après avoir lu cet article, vous connaissez les différents types de relations et savez comment ils contribuent à fournir une solution de modélisation pratique dans les bases de données relationnelles.