

Bases de données

Stockage : Clé primaire, clé étrangère et relations entre tables

- Comment choisir la clé d'une table ?
- Quelle est la différence entre clé primaire et clé étrangère ?
- Comment relier les informations contenues dans différentes tables d'une même base de données ?

1. Clé primaire

Une **clé primaire** est formée d'un ou plusieurs **champs** d'une base de données qui identifient de manière unique un enregistrement dans une table. Elle permet ainsi d'éviter les doublons. Par exemple, le numéro d'étudiant permet d'identifier un étudiant de manière unique.

Une clé primaire peut être un code en partie compréhensible comme le numéro de sécurité sociale ou être un **champ numérique auto-incrémenté**, c'est-à-dire que les enregistrements sont numérotés automatiquement dans la table. Le premier enregistrement aura le numéro 1 et ainsi de suite.

Prenons l'exemple de la base de données textuelles qui se trouve dans la section 4 de cette fiche. Cette base de données, composée de 4 tables (Document, Phrase, Index et Mot), permet la recherche de mots dans un ensemble de documents et la visualisation des phrases dans lesquelles ces mots apparaissent. Les champs *Num_mot* de la table MOT, *Num_phrase* de la table PHRASE et *Num_doc* de la table DOCUMENT sont des clés primaires numériques auto-incrémentées.

Une clé primaire peut correspondre à l'association de plusieurs champs dans une table, comme c'est le cas dans la table INDEX, dont la clé est formée par le couple des champs *Mot* et *Phrase* : on parle alors de **clé composée**. Dans ce cas, il faut que chaque combinaison des champs de la clé soit unique.

Notation : par convention, les champs composant la clé primaire sont soulignés.

2. Clé étrangère

Outre la préservation de l'unicité des enregistrements, les clés servent également à établir des **relations** entre les tables. Pour cela, on utilise des **clés étrangères**, c'est-à-dire des clés qui représentent la valeur de la clé primaire d'une table connexe. Contrairement aux clés primaires, qui doivent être uniques, les clés étrangères peuvent prendre plusieurs fois la même valeur dans une table. Par exemple, le numéro d'un document (*Num_doc*) est la clé primaire de la table DOCUMENT et est utilisée comme clé étrangère dans la table PHRASE (champ *Doc_source*). Comme un document contient généralement plusieurs phrases, on pourra retrouver plusieurs fois la même valeur de numéro de document dans le champ *Doc_source* de la table PHRASE.

Notation : par convention, la clé étrangère est précédée du caractère # (parfois, elle se trouve également en italique)

3. Types de relations entre tables

1 à plusieurs (¥) : un enregistrement d'une table est en relation avec plusieurs enregistrements d'une autre table. Par exemple, un document comporte plusieurs phrases mais une phrase ne peut appartenir qu'à un seul document.

🚲 1 à 1 : un enregistrement d'une table est en relation avec un seul enregistrement d'une autre table. Prenons l'exemple de 2 tables : SECURITE_SOCIALE (Num_Secu, Nom Prenom), ETUDIANT (No_Etudiant, No_INSEE), reliées par le champ indiquant le numéro de sécurité sociale (respectivement *num_secu* et *No_INSEE*). Puisqu'un étudiant a un seul numéro de sécurité sociale, un enregistrement de la table ETUDIANT est en relation avec un seul enregistrement de la table SECURITE_SOCIALE et inversement.

🚲 **plusieurs à plusieurs** : dans ce cas, il est nécessaire de créer une table intermédiaire comme la table INDEX. En effet, un mot peut appartenir à plusieurs phrases différentes et une phrase contient plusieurs mots. Il est impossible de savoir à priori le nombre de mots contenus dans une phrase, et, inversement, le nombre de phrases dans lesquelles se trouve un mot. La seule solution est de créer une nouvelle table, dans laquelle il est possible d'ajouter autant de couples "mot-phrase" que nécessaire.

Intégrité référentielle : Lors de la création d'une relation, il est très fortement conseillé de lui appliquer l'intégrité référentielle. Il s'agit d'un ensemble de contrôles que le SGBD effectue, pour assurer la cohérence interne de la BD. Par exemple, si nous appliquons l'intégrité référentielle à la relation entre les tables DOCUMENTS et PHRASES, un numéro de document ne figurant pas dans la table DOCUMENTS sera refusé dans la table PHRASES et il ne sera pas possible de supprimer un document de la table DOCUMENTS si son numéro est utilisé dans la table PHRASES.

4. Exemple : base de données textuelles

