

```
ALTER INDEX ALL ON nom_table REBUILD;
```

DML : Insertion d'enregistrements dans une table

Il existe deux manières d'insérer des données dans une table :

- La première consiste à spécifier le nom des colonnes et les valeurs correspondantes. Dans ce cas, il faut veiller au fait que toutes les colonnes non nulles soient mentionnées.

```
INSERT INTO nom_table ( nom_colonne_3 , nom_colonne_1 ) VALUES ( valeur_3 , valeur_1 );
```

- La seconde consiste à insérer une valeur pour toutes les colonnes. Dans ce cas, il faut veiller à bien respecter l'ordre des colonnes dans la table.

```
INSERT INTO nom_table VALUES ( valeur_1, valeur_2 , valeur_3, ... );
```

Lorsque vous souhaitez insérer des enregistrements dans une table ayant une colonne auto-incrémentée, préférez la première méthode en omettant de spécifier la colonne auto-incrémentée et sa valeur.

Attention à l'ordre dans lequel vous insérez vos données dans vos tables, en particulier si celles-ci comportent des clés étrangères. Si votre enregistrement fait référence à une valeur contenue dans un champs d'une autre table, il faut que cette valeur soit déjà enregistrée dans la seconde table.

DML : Modification des enregistrements contenus dans une table

Requête permettant de modifier un champ de tous les enregistrements déjà insérés dans une table.

```
UPDATE nom_table  
SET colonne_3 = valeur_3;
```

Requête permettant de modifier plusieurs champs de tous les enregistrements déjà insérés dans une table.

```
UPDATE nom_table  
SET colonne_3 = valeur_3, colonne_1 = valeur_1;
```

Requête permettant de modifier un champ des enregistrements déjà insérés dans une table avec une condition.

```
UPDATE nom_table  
SET colonne_1 = valeur_1  
WHERE condition;
```

Requête permettant de modifier plusieurs champs des enregistrements déjà insérés dans une table avec une condition.

```
UPDATE nom_table  
SET colonne_1 = valeur_1 , colonne_3 = valeur_3  
WHERE condition;
```

Attention à l'ordre dans lequel vous modifiez vos données dans vos tables, en particulier si celles-ci comportent des clés étrangères. Si votre enregistrement fait référence à une valeur contenue dans un champs d'une autre table, il faut que cette valeur existe déjà dans la seconde table.

DML : Suppression des enregistrements contenus dans une table

Requête permettant de supprimer tous les enregistrements d'une table.

```
DELETE FROM nom_table;
```

Requête permettant de supprimer tous les enregistrements d'une table satisfaisant une condition.

```
DELETE FROM nom_table  
WHERE condition;
```

Attention à l'ordre dans lequel vous supprimez les enregistrements, en particulier si vos tables comportent des clés étrangères. Si l'enregistrement que vous souhaitez supprimer est référencé par une valeur contenue dans un champs d'une autre table, il faut d'abord supprimer cette valeur de la seconde table.