

III / Purger une table

a) Syntaxe

```
TRUNCATE TABLE `tbl_nom` ;
```

b) Exemple

```
TRUNCATE TABLE `ma_table` ;
```

Cette requête supprimera toutes les données de la table nommée ``ma_table``.

Si la table contient un attribut avec `AUTO_INCREMENT`, la valeur de la prochaine incrémentation sera réinitialisée à 1.

III / Les jointures

a) Qu'est ce que c'est ?

Les jointures permettent d'associer plusieurs tables dans une seule requête.

b) Exemple

Je souhaite afficher les attributs ``login``, ``email`` de la table ``compte`` ainsi que les attributs ``prenom``, ``nom`` et ``genre`` de la table ``infos_persos`` où l'attribut ``compte`.`id`` = ``infos_persos`.`id_compte``.

login	email	prenom	nom	genre
john89	John.coffey89@gmail.com	John	Coffey	homme
john_89	john_hurt@gmail.com	John	Hurt	homme
eric75	eric_bana_75@gmail.com	Eric	Bana	homme

III/ INNER JOIN

a) Description

INNER JOIN retourne les données où il y a au moins une colonne dans chaque table qui correspond à la condition.

b) Syntaxe

```
SELECT * FROM `tbl_nom1`
```

```
INNER JOIN `tbl_nom2`
```

```
ON `tbl_nom1`.`col_nom` = `tbl_nom2`.`col_nom`
```

III/ LEFT JOIN

a) Description

LEFT JOIN retourne toutes les données de la table de gauche même si ça ne correspond pas à la condition.

b) Syntaxe

```
SELECT * FROM `tbl_nom1`
```

```
LEFT JOIN `tbl_nom2`
```

```
ON `tbl_nom1`.`col_nom` = `tbl_nom2`.`col_nom`
```

III/ RIGHT JOIN

a) Description

RIGHT JOIN retourne toutes les données de la table de droite même si ça ne correspond pas à la condition.

b) Syntaxe

```
SELECT * FROM `tbl_nom1`
```

```
RIGHT JOIN `tbl_nom2`
```

```
ON `tbl_nom1`.`col_nom` = `tbl_nom2`.`col_nom`
```

III/ FULL JOIN

a) Description

FULL JOIN retourne toutes les données des deux tables même si ça ne correspond pas à la condition.

b) Syntaxe

```
SELECT * FROM `tbl_nom1`  
RIGHT JOIN `tbl_nom2`  
ON `tbl_nom1`.`col_nom` = `tbl_nom2`.`col_nom`
```

III/ NATURAL JOIN

a) Description

NATURAL JOIN retourne toutes les données des deux tables s'il y a une colonne du même nom et du même type dans chaque table.

b) Syntaxe

```
SELECT * FROM `tbl_nom1`  
NATURAL JOIN `tbl_nom2`
```

III/ Jointure associative

a) Description

Une jointure associative join un minimum de deux table sans utiliser le mot clé **JOIN** et en spécifiant la condition dans **WHERE**.

b) Syntaxe

```
SELECT *  
FROM `tbl_nom1`, `tbl_nom2` [, `tbl_nom3`]  
WHERE `tbl_nom1`.`col_nom` = `tbl_nom2`.`col_nom`  
[{AND | OR} `tbl_nom1`.`col_nom` = `tbl_nom3`.`col_nom`]
```

c) AND | OR

AND (et) | **OR** (ou) permet de spécifier plusieurs condition.

III/ Modifier une table

a) Renommer un attribut.

```
ALTER TABLE `tbl_nom`
```

```
CHANGE col_nom col_nouveau_nom type definition
```

b) Modifier un attribut

```
ALTER TABLE `tbl_nom`
```

```
CHANGE col_nom type definition
```

c) Information

Quand on modifie un attribut, il faut toujours préciser sa définition complète. (`NOT NULL`, `NULL`, `DEFAULT`, `AUTO_INCREMENT`, `COMMENT`)