

SQL Mode

Le serveur MySQL peut fonctionner dans différents modes SQL et peut appliquer ces modes différemment pour différents clients, en fonction de la valeur de la variable système `sql_mode`. Les administrateurs de base de données peuvent définir le mode SQL global pour répondre aux exigences opérationnelles du serveur de site. Chaque application peut définir son mode de session SQL en fonction de ses propres exigences.

- `ALLOW_INVALID_DATES`

N'effectuez pas de vérification complète des dates. Vérifiez uniquement que le mois est compris entre 1 et 12 et le jour entre 1 et 31.

- `ANSI_QUOTES`

Traitez "comme un caractère de citation d'identificateur (comme le ` caractère de citation) et non comme un caractère de citation de chaîne.

- `ERROR_FOR_DIVISION_BY_ZERO`

Le mode `ERROR_FOR_DIVISION_BY_ZERO` affecte le traitement de la division par zéro, ce qui inclut `MOD (N, 0)`. Pour les opérations de changement de données (`INSERT`, `UPDATE`), son effet dépend également de l'activation du mode SQL strict.

- - Si ce mode n'est pas activé, la division par zéro insère `NULL` et ne produit aucun avertissement.
 - Si ce mode est activé, la division par zéro insère `NULL` et génère un avertissement.

- `HIGH_NOT_PRECEDENCE`

Définiie la précedence de l'opérateur `NOT` .

- `NO_BACKSLASH_ESCAPES`

Désactive l'utilisation du caractère barre oblique inverse (`\`) en tant que caractère d'échappement dans les chaînes. Lorsque ce mode est activé, la barre oblique inverse devient un caractère ordinaire comme un autre.

- `NO_ENGINE_SUBSTITUTION`

Contrôle la substitution automatique du moteur de stockage par défaut lorsqu'une instruction telle que CREATE TABLE ou ALTER TABLE spécifie un moteur de stockage désactivé ou non compilé.

- ANSI

Équivalent à REAL_AS_FLOAT, PIPES_AS_CONCAT, ANSI_QUOTES, IGNORE_SPACE et ONLY_FULL_GROUP_BY.

- TRADITIONAL

Equivalent à STRICT_TRANS_TABLES, STRICT_ALL_TABLES, NO_ZERO_IN_DATE, NO_ZERO_DATE, ERROR_FOR_DIVISION_BY_ZERO et NO_ENGINE_SUBSTITUTION.

Configuration conseillé

Dans le fichier my.ini initialisé sql_mode a ANSI et TRADITIONAL. Ces parametres renforce la sécurité et une proximité avec Oracle <source lang='ini'> sql_mode = "ANSI,TRADITIONAL" </source>

Une fois redemarré, le nouveaux SQL mode des clients est homogène à <source lang='SQL'> </source>

```
+-----+
+-----+
-----+
| Variable_name | Value | +-----+
+-----+
-----+
| sql_mode |
REAL_AS_FLOAT,PIPES_AS_CONCAT,ANSI_QUOTES,IGNORE_SPACE,ONLY_FULL_GROUP_BY,ANSI,STRICT_TRANS_TA
BLES,STRICT_ALL_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,TRADITIONAL,NO_
ENGINE_SUBSTITUTION |
+-----+
+-----+
-----+ 1 row in set (0,04 sec)
```

Test SQL Mode

la variable `sql_mode` permet de savoir le mode sql par default du serveur

```
mysql> SHOW SESSION VARIABLES LIKE 'sql_mode';
+-----+
+-----+
-----+
| Variable_name | Value |
+-----+
+-----+
-----+
| sql_mode |
ONLY_FULL_GROUP_BY, STRICT_TRANS_TABLES, NO_ZERO_IN_DATE, NO_ZERO_DATE, ERROR_FOR_DIVISION_BY_ZERO
, NO_ENGINE_SUBSTITUTION |
+-----+
+-----+
-----+
1 row in set (0,00 sec)
```

Il est possible de tester le mode et d'override pour la session la prise en compte du SQL

```
mysql> create table test(i INT); Query OK, 0 rows affected (0,12 sec)

mysql> insert into test values (2/0); ERROR 1365 (22012): Division by 0

mysql> set sql_mode=; Query OK, 0 rows affected (0,00 sec)

mysql> insert into test values (2/0); Query OK, 1 row affected (0,06 sec)
```

Revision #2

Created 23 November 2019 15:49:12 by Admin

Updated 26 November 2019 19:22:54 by Admin