

Manipulação de Erros em MySQL

A implementação de um manipulador de exceção de uso geral seria uma boa prática, salvo que se você não pode revelar a razão pela qual a exceção ocorreu, você torna a depuração das suas procedures difícil ou impossível. Por exemplo, considere:

```
DECLARE CONTINUE HANDLER FOR SQLEXCEPTION
BEGIN
    SET I_status=-1;
    Set I_message='Algum tipo de erro detectado na aplicação';
END;
```

Receber uma mensagem de erro como esta não é de muita ajuda. Ocultar a causa real do erro faz com que o manipulador de condição seja inútil na maioria das circunstâncias.

A partir do SQL 2003 é possível o acesso direto aos valores de SQLCODE e SQLSTATE. Veja alguns destes valores no texto a seguir.

SQLSTATE ou MySQL Código de erro ?

Em teoria, usar os códigos SQLSTATE vai fazer o seu código mais portátil para outro banco de dados e poderia, portanto, parece ser a melhor escolha. No entanto, há um número de razões para usar os códigos de erro do MySQL, em vez de códigos SQLSTATE ao escrever MySQL *stored procedures*:

- Na realidade, é improvável que você vá mover suas *stored procedures* para outro SGBD. As *stored procedures* Oracle e SQL Server são totalmente incompatíveis com o MySQL. A linguagem de *stored procedures* do DB2 é um pouco compatível (ambos são baseados na SQL 2003).
- Nem todos os códigos de erro do MySQL têm SQLSTATE equivalentes. Embora cada código de erro MySQL seja associado a algum SQLSTATE, muitas vezes você vai encontrar um SQLSTATE de propósito geral, que não é específico (como HY000). Portanto, você certamente vai ter que codificar alguns manipuladores que se referem diretamente ao código de erro MySQL.

Tabela 1 – Resumo dos códigos de erros e SQLStates do MySQL.

Error code	SQLState	Error message
1011	HY000	Error on delete of '%s' (errno: %d)
1021	HY000	Disk full (%s); waiting for someone to free some space...
1022	23000	Can't write; duplicate key in table '%s'
1027	HY000	'%s' is locked against change
1036	HY000	Table '%s' is read only
1048	23000	Column '%s' cannot be null
1062	23000	Duplicate entry '%s' for key %d
1099	HY000	Table '%s' was locked with a READ lock and can't be updated
1100	HY000	Table '%s' was not locked with LOCK TABLES
1104	42000	The SELECT would examine more than MAX_JOIN_SIZE rows; check your WHERE and use SET SQL_BIG_SELECTS=1 or SET SQL_MAX_JOIN_SIZE=# if the SELECT is okay
1106	42000	Incorrect parameters to procedure '%s'
1114	HY000	The table '%s' is full
1150	HY000	Delayed insert thread couldn't get requested lock for table %s
1165	HY000	INSERT DELAYED can't be used with table '%s' because it is locked with LOCK TABLES
1242	21000	Subquery returns more than 1 row
1263	22004	Column set to default value; NULL supplied to NOT NULL column '%s' at row %ld

Error code	SQLState	Error message
1264	22003	Out of range value adjusted for column '%s' at row %ld
1265	1000	Data truncated for column '%s' at row %ld
1312	0A000	SELECT in a stored program must have INTO
1317	70100	Query execution was interrupted
1319	42000	Undefined CONDITION: %s
1325	24000	Cursor is already open
1326	24000	Cursor is not open
1328	HY000	Incorrect number of FETCH variables
1329	2000	No data to FETCH
1336	42000	USE is not allowed in a stored program
1337	42000	Variable or condition declaration after cursor or handler declaration
1338	42000	Cursor declaration after handler declaration
1339	20000	Case not found for CASE statement
1348	HY000	Column '%s' is not updatable
1357	HY000	Can't drop a %s from within another stored routine
1358	HY000	GOTO is not allowed in a stored program handler
1362	HY000	Updating of %s row is not allowed in %s trigger
1363	HY000	There is no %s row in %s trigger

Exemplos

```

DECLARE sqlcode INT DEFAULT 0;
DECLARE status_message VARCHAR(50);
DECLARE CONTINUE HANDLER FOR duplicate_key
BEGIN
    SET sqlcode=1062;
    SET status_message='Duplicidade de chave primária';
END;
DECLARE CONTINUE HANDLER FOR foreign_key_violated
BEGIN
    SET sqlcode=1216;

    SET status_message='Violação de chave estrangeira';
END;
DECLARE CONTINUE HANDLER FOR NOT FOUND
BEGIN
    SET sqlcode=1328;
    SET status_message='Não mais dados';
END;

```

Veja relação completa dos códigos de erro MySQL em: <http://www.briandunning.com/error-codes/?source=MySQL>