平时我们在设计数据库表的时候总会设计 unique  或者 给表加上 primary key 的限制条件.

此时 插入数据的时候 ，经常会有这样的情况：

我们想向数据库插入一条记录：

 若数据表中存在以相同主键的记录，我们就更新该条记录。

否则就插入一条新的记录。

最原始的办法：先查，如果存在，先删除再插入，如果不存在，直接删除。

但是这样写有两个问题

1、效率太差，每次执行都要执行2个sql  
2、高并发的情况下数据会出问题，不能保证原子性

**ON DUPLICATE KEY UPDATE**达到以上目的, 且能保证操作的原子性和数据的完整性。

ON DUPLICATE KEY UPDATE 可以达到以下目的:

向数据库中插入一条记录：

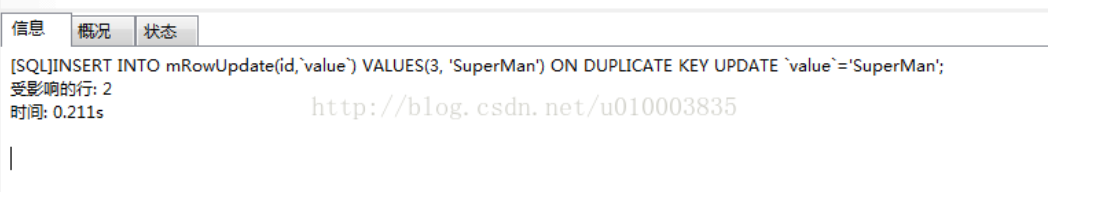
若该数据的主键值/ UNIQUE KEY 已经在表中存在,**则执行更新操作, 即UPDATE 后面的操作。**

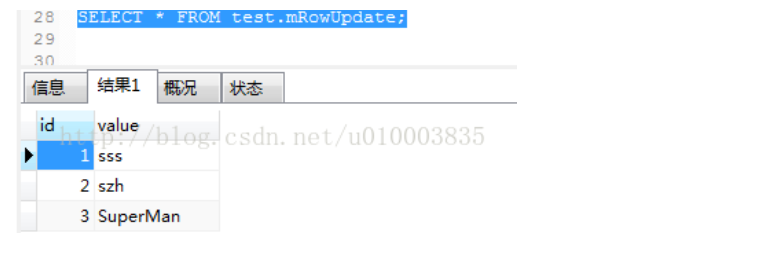
否则插入一条新的记录。

DROP TABLE IF EXISTS `mRowUpdate`;  
CREATE TABLE `mRowUpdate` (  
  `id` int(11) NOT NULL,  
  `value` varchar(255) DEFAULT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB;

INSERT INTO `mRowUpdate` VALUES ('1', 'sss');  
INSERT INTO `mRowUpdate` VALUES ('2', 'szh');  
INSERT INTO `mRowUpdate` VALUES ('3', '9999');

INSERT INTO mRowUpdate(id,`value`) VALUES(3, 'SuperMan') ON DUPLICATE KEY UPDATE `value`='SuperMan';





ON DUPLICATE KEY UPDATE 特别适用于多行插入。如:

INSERT INTO `table` (`a`, `b`, `c`) VALUES (1, 2, 3), (4, 5, 6) ON DUPLICATE KEY UPDATE `c`=VALUES(`a`)+VALUES(`b`);

  Tips: VALUES()函数只在INSERT…UPDATE语句中有意义，其它时候会返回NULL。