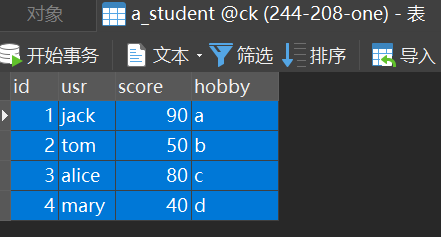
要更新多行数据的多个字段的值，且不同行的同一个字段，更新为不同的值；这种就是常用的更新sql无法做到的，因为常用的更新sql都是将多行数据的相同字段变为同样的值。

前情数据：



执行：

UPDATE a\_student SET

score=CASE id

WHEN 2 THEN 60

WHEN 4 THEN 60

END,

hobby=CASE usr//这里usr 也可仍是id，但不可是score，因上一个CASE是针对score

WHEN 'tom' THEN 't-no'

WHEN 'mary' THEN 'm-no'

END WHERE id IN(2,4) AND usr IN('tom', 'mary');

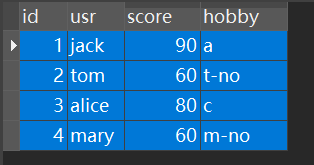
//注意： 两个CASE在更新表中数据前需要先各自定位到表中的行，这个定位必须通过在WHERE后面的两个IN来各自实现的，因此各自CASE的IN必须写，但如果两个IN括号内的内容相同，则只需写一个。

然而，由于两个IN又是作为一个整体来给整个更新sql语句进行定位，因此**两个IN括号中的值代表的行的集合，必须相同，而且两个IN之间只能用AND而不能是OR来连接**。

否则，当没有写IN(2,,4)时，会将id（usr类似的同理）不是2和4的所有的score置空；

否则，当两个IN之间不是用的AND而是用的OR，而两个IN括号内的值代表的行的集合又不相同时，会将id不是2和4的其他的某行（取决于另一个IN定位到了除了2和4之外的行）的score置空。若建表时score不为NULL，则会报错说1048 - Column 'score' cannot be null。

否则，当两个IN之间用的是正确的AND，但两个IN括号内的值代表的行的集合又不相同时，会导致最终只更新了 第一个IN括号内的值代表的行 和 第二个IN括号内的值代表的行 中相同的那些行。



=========================

//注意： 切记，**第二组的usr处不能是第一组的=CASE前面的那个score**！ 可能是因为本次被更改的列不能同时作为更改另一列的定位依据。不信可见下面的错误示例：

比如执行：

UPDATE a\_student SET

score=CASE id

WHEN 2 THEN 60

when 4 THEN 60

END,

hobby=CASE score

WHEN 90 THEN 'x'

WHEN 80 THEN 'xx'

END WHERE id IN(2,4) AND score IN(90,80);



或者比如执行：

UPDATE a\_student SET

score=CASE id

WHEN 2 THEN 60

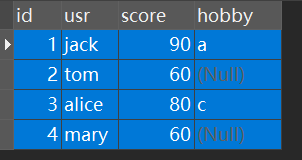
when 4 THEN 60

END,

hobby=CASE score

WHEN 40 THEN 'x'

WHEN 50 THEN 'xx'

END WHERE id IN(2,4) AND score IN(40,50);（结果不符合预期）

如果建表时hobby字段不为NULL，则会报错1048 - Column 'hobby' cannot be null

或者比如执行：

UPDATE a\_student SET

score=CASE id

WHEN 2 THEN 60

when 4 THEN 60

END,

hobby=CASE score

WHEN 60 THEN 'x'

WHEN 60 THEN 'xx'

END WHERE id IN(2,4) AND score IN(60,60);



（完）