

# Insert语句

## 语法



insert into 表名 (字段1,字段2,字段3..) values(值1,值2,值3...);

有几列就插入多少的值(一一对应)。

insert into 表名 values(值1,值2,值3...); 插入所有的列

注意：

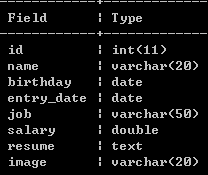
1.数据与字段的类型相同。

2.字段长度需要控制。

3.字符串或者日期类型需要使用' '

## 练习：

使用insert语句向表中插入三个员工的信息。



insert into user values (1,'xiaofeng','1994-10-10','2011-1-1','HR',19000,'aaa','abc');

nsert into user values (2,'美美','1994-10-10','2011-1-1','HR',19000,'aaa','abc');

insert into user values (3,'小风','1994-10-10','2011-1-1','WORKER',21000,'aaa','abc');



## 解决中文乱码

插入一条中文记录

insert into user values (2,'美美','1994-10-10','2011-1-1','HR',19000,'aaa','abc');

出错了：

ERROR 1366 (HY000): Incorrect string value: '\xC3\xF7' for column 'name' at row 1 ;

错误原因：

mysql client 采用默认字符集编码 gbk

查看系统所有字符集 ：

show variables like 'character%';

MYSQL中共有6个地方字符集 ：client connetion result 和客户端相关 、database server system 和服务器端相关

1.client是客户端使用的字符集。

2.connection是连接数据库的字符集设置类型，如果程序没有指明连接数据库使用的字符集类型就按照服务器端默认的字符集设置。

3.database是数据库服务器中某个库使用的字符集设定，如果建库时没有指明，将使用服务器安装时指定的字符集设置。

4.results是数据库给客户端返回时使用的字符集设定，如果没有指明，使用服务器默认的字符集。

5.server是服务器安装时指定的默认字符集设定。

6.system是数据库系统使用的字符集设定。（utf-8不可修改）

解决办法：

修改客户端字符集为gbk

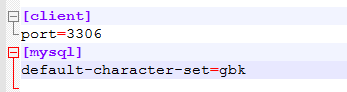
第一种：当前窗口临时修改 set names gbk ;

\* 只对当前窗口有效，关闭后就会失效

第二种：配置mysql/my.ini 文件

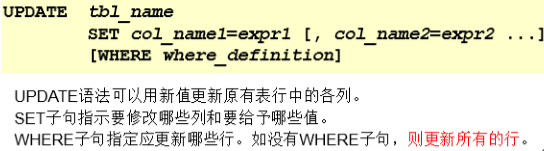
[mysql] 客户端配置

[mysqld] 服务器端配置



# Update语句

## 语法



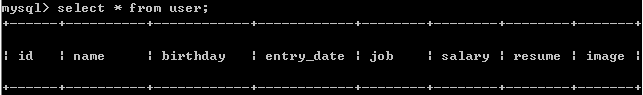
语法：

update 表名 set 字段=值,字段=值... [where ]

\* 如果没有where条件，默认更新所有的记录。

\* 有where提交，选择某一条记录。

## 练习



将所有员工薪水修改为5000元。

update user set salary=5000;

将姓名为’班长’的员工薪水修改为3000元。

update user set salary=3000 where name='班长';

将姓名为’美美’的员工薪水修改为4000元,job改为BOSS。

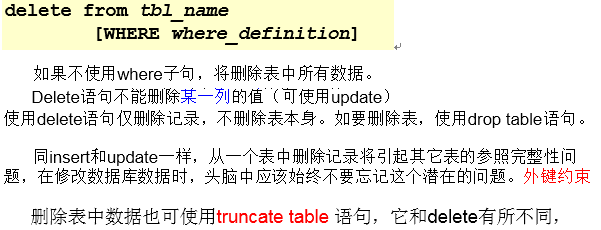
update user set salary=4000,job='BOSS' where name='美美';

将班长的薪水在原有基础上增加1000元。

update user set salary = salary+1000 where name='班长';

# Delete语句

## 语法



语法：

delete from 表名 [where ]; 删除数据

truncate 表名; 删除所有的数据

## truncate 和 delete的区别：

truncate删除数据，先删除整个表。再创建一个新的空的表。（效率）

delete删除数据，一条一条删除的。（\*\*\*\*\*）

事物（insert update delete）

## 练习

删除表中名称为’班长’的记录。

\* delete from user where username='班长';

删除表中所有记录。

delete from user;

使用truncate删除表中记录。

\* truncate user;

# 查询

## 查询插入

### 第一种

INSERT INTO 表X(列A,列B,列C…)[SELECT](https://www.baidu.com/s?wd=SELECT&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1d9nj9hPH79PW99uHN9nyRY0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3En1f4Pjm4n1T4" \t "_blank) 列1,列2,列3… [FROM](https://www.baidu.com/s?wd=FROM&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1d9nj9hPH79PW99uHN9nyRY0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3En1f4Pjm4n1T4" \t "_blank) 表B [WHERE](https://www.baidu.com/s?wd=WHERE&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1d9nj9hPH79PW99uHN9nyRY0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3En1f4Pjm4n1T4" \t "_blank)条件 注意，查询[结果集](https://www.baidu.com/s?wd=%E7%BB%93%E6%9E%9C%E9%9B%86&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1d9nj9hPH79PW99uHN9nyRY0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3En1f4Pjm4n1T4" \t "_blank)： 对应列的数据类型要和插入的列的一样。 列的数目也要一样。  
如果两张表的结构一样，直接  
INSERT INTO 表A [SELECT](https://www.baidu.com/s?wd=SELECT&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1d9nj9hPH79PW99uHN9nyRY0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3En1f4Pjm4n1T4" \t "_blank) \* [FROM](https://www.baidu.com/s?wd=FROM&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1d9nj9hPH79PW99uHN9nyRY0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3En1f4Pjm4n1T4" \t "_blank) 表B

### 第二种

**update KUN\_API\_PERMIT u SET u.USER\_TIMES\_LIMIT\_USED =**

**(select REQ\_TIMES from (SELECT API\_ID AS API\_ID, APP\_KEY AS APP\_KEY, SUM(REQ\_TIMES) AS REQ\_TIMES FROM kun\_api\_cdr\_total GROUP BY API\_ID, APP\_KEY) T where T.API\_ID = u.API\_ID AND T.APP\_KEY = u.APP\_KEY );**

## 查询更新

必须使用内连接

**UPDATE 表 INNER JOIN**

**( 更新表源：带id及更新数据 ) T ON 连接**

**SET 更新字段， 更新字段， 更新字段，**

