

一. 基础知识

I. AUTO_INCREMENT

1. 存在意义：在新加一个元素的时候，使用最简单的编号——下一个编号。
2. 覆盖：有时候我们需要给特殊的元素特殊的编号。这时候我们可以用INSERT INTO来指定编号，注意其后面所有新增的元素编号都得从该编号开始。
3. 缺点：使用自动增量时，我们并不知道这些值是什么。这个时候，可以用SELECT last_insert_id () 来获取该编号的值。
4. 其它：每一个表只允许一个AUTO_INCREMENT列。

II. DEFAULT

有些时候没有给出值，但是我们可以赋给它默认值。

III. ENGINE

1. 引擎的作用：创建表，处理你的SELECT等请求。
2. 关于mysql：mysql有多个引擎，他们分别藏在mysql服务器内。
3. 引擎分类：
 - ①InnoDB：可靠的事务处理引擎，但不支持全文本搜索。
 - ②MyISAM：性能极高，支持全文本搜索，但不支持事务处理。
 - ③MEMORY：功能上等同于MyISAM，将数据存储在内存，所以速度很快。

二. 创建表

1. 一些格式：

①CREATE TABLE *tablename* (...)

②在()里面，给出至少三个信息：列名，类型，是否允许NULL（还有是否是自动增量，是否有默认值）

③在()的最后，给出PRIMARY KEY (*columnname*)

④在()的后面，给出ENGINE = ... ;

2. 使用示例：mysql> CREATE TABLE 409_student (

-> stu_id int NOT NULL AUTO_INCREMENT,

-> stu_sex char(4) NOT NULL DEFAULT 'male',

-> stu_gpa float NULL,

-> PRIMARY KEY(stu_id)

->)ENGINE = InnoDB;

3. 格式提要：

①每行各个字符串之间用tab键分隔

②类型（如上面的int, char(n), float）均使用小写

③注意定义一列要使用一行，最后用逗号（,）结束。

二. 删除表

想要彻底删除一个表格十分地简单，使用下面的语句即可：

```
DROP TABLE tablename;
```

三. 重命名表

这个也很简单：

```
RENAME TABLE tablename TO new_tablename;
```

四. 更新表

为更新表的定义，可以使用ALTER TABLE语句（理想的情况下，表在定义后就不用再更新了）

所有的表的更新，都以ALTER TABLE *tablename*开头

1. 增加列：ADD *columnname type*;
2. 删除列：DROP COLUMN *columnname*;