Data Definition Language——数据定义语言

DDL是对数据内部的对象进行创建，删除，修改等的操作语言。

与DML语言的区别：DML只是对表内部数据的操作，而不涉及到表的定义，结构的修改，更不会涉及到其它对象。

1. 连接数据库：mysql -uroot -p;

命令的结束符用;或者\g结束。

1. 创建数据库：create database dbname;

查看系统中存在哪些数据库：show databses;

information\_schema：主要存储了系统中的一些数据库对象信息。比如用户表信息、列信信息、权限信息、字符集信息、分区信息等。  
cluster：存储了系统的集群信息。  
mysql：存储了系统的用户权限信息。  
test：系统自动创建的测试数据库，任何用户都可以使用。

选择要操作的数据库：use dbname;

1. 删除数据库（所有表数据都会被删除）：drop database dbname;
2. 创建表：create table tablename(列名 列的数据类型 列的约束条件, 列名 列的数据类型 列的约束条件);

查看表的定义，可以使用desc tablename;

查看全面的表定义信息，有时需要通过查看创建表的SQL语句：

show create table emp \G;

1. 删除表：drop table tablename;
2. 修改表（多数情况下表结构的修改用alter table语句）：
   1. 修改表类型：alter table tablename modify [column] column\_definition [first | after col\_name];

修改表emp中ename字段定义，将varchar(10)改为varchar(20)：alter table emp modify ename varchar(20);

* 1. 增加表字段：alter table tablename add [column] column\_definition [first | after col\_name];

表emp上新增加字段age，类型为int(3)：alter table emp add age int(3);

* 1. 删除表字段：alter table tablename drop [column] col\_name;

将表emp上的字段删除掉：alter table emp drop age;

* 1. 给字段改名：alter table tablename change [column] old\_col\_name column\_definition [first | after col\_name];

将字段age改名为age1，同时修改字段类型为int(4)：alter table emp change age age1 int(4);

* 1. 修改字段的排列顺序：

将新增字段brith date加在ename之后：alter table emp brith date after ename;

注意：CHANGE/FIRST|AFTER COLUMN 这些关键字都属于MySQL，在标准 SQL 上的扩展，在其他数据库上不一定适用。

* 1. 表改名：alter table tablename rename [to] new\_tablename;

将表emp改名为emp1：alter table emp rename emp1;