Mysql数据库操作

mysql结构化查询语言

结构化查询语言sql包含四个部分:

1.DDL　//数据定义语言,create,drop,alter

2.DML　//数据操作语言,insert,update,delete

3.DQL　//数据查询语言,select

4.DCL　//数据控制语言,grant,commit,rollback

insert

- insert into user(name) values("user4");

update

- update user set name="user4" where id=4;
- update user set name='user5',age=20 where id=5;

delete

- delete from user where id>=3 and id<=5;
- delete from user where id between 3 and 5;
- delete from user where id=3;
- delete from user where id in(1,3,5);

select

1.选择特定的字段

```
select id,name from user;
```

2.给字段取别名-as

```
select id,name from user;
select id maoxian,name from user;
select id as maoxian,name from user;
```

3.distinct关键字的使用

```
select distinct age from user;
```

4.使用where条件进行查询

```
select * from user where id>=3 and id<=5;
```

5.查询空值null

```
select * from user where age is null;
select * from user where age is not null;
```

6.between and的使用方法

```
select * from user where id between 3 and 5;
```

7.in的使用方法

```
select * from user where id=1 or id=2 or id=10;
select * from user where id in(1,2,10);
```

建议使用这个

8.like的使用方法

% 匹配所有

_ 匹配一个字符

```
select * from user where  name like "%mysql%";
   //%在前，name这一列的索引会失效
```

9.使用order by对查询结果排序

```
select * from user order by id asc;  //默认就是升序,数字从小到大
select * from user order by id desc;  //默认就是升序,数字从大到小
```

10.使用limit限定输出个数(分页实现)

```
select * from user order by id limit 0,2;
select * from user order by id limit 5; //limit 0,5 前五个
```

mysql常用函数-select

- 连接函数concat()

```
select concat(id,age) from user;
select concat("aaa","bbb","cccc");
```

- 随机数rand()
- select * from user order by rand() limit 3;

mysql常用函数-select

- 统计个数count()

```
select count(id) from user;
```

- 统计表总行数

```
select count(*) from user;
```

mysql常用函数-select

- 求和sum()

```
select sum(age) from user;
```

- 平均值avg()

```
select avg(age) from user;
```

- 最大值max()

select max(age) from user;

- 最小值min()
- select min(age) from user;

group by分组聚合的使用

统计每个班人数和总成绩

```
mysql> select clsnum,count(id),sum(id) from classtab
    group by clsnum;
+--------+-----------+---------+
| clsnum | count(id) | sum(id) |
+--------+-----------+---------+
|      1 |         3 |       6 |
|      2 |         1 |       4 |
|      3 |         2 |      11 |
+--------+-----------+---------+
3 rows in set (0.00 sec)
```

group by分组聚合的使用

分组聚合加条件

```
mysql> select clsnum,count(id),sum(id) from classtab
    group by clsnum having sum(
id)>=5;
+--------+-----------+---------+
| clsnum | count(id) | sum(id) |
+--------+-----------+---------+
|      1 |         3 |       6 |
|      3 |         2 |      11 |
+--------+-----------+---------+
2 rows in set (0.00 sec)
```

多表查询

有一种下面三种方法都能实现，优先选择普通多表查询

1.普通多表查询

```
mysql> select classtab.id,classtab.name,classtab.clsnum,score.num from
    classtab,
score where classtab.id=score.cid;
+----+-------+--------+------+
| id | name  | clsnum | num  |
+----+-------+--------+------+
|  1 | user1 |      1 |   70 |
|  3 | user3 |      1 |   59 |
|  5 | user5 |      3 |   80 |
+----+-------+--------+------+
3 rows in set (0.00 sec)
```

嵌套查询|子查询-in

```
mysql> select name from classtab where id
    in(select cid from score);
+-------+
| name  |
+-------+
| user1 |
| user3 |
| user5 |
+-------+
3 rows in set (0.03 sec)
```

左链接查询-left join on

```
mysql> select classtab.id,classtab.name,classtab.clsnum,score.num
    from classtab
left join score on classtab.id=score.cid;
+----+-------+--------+------+
| id | name  | clsnum | num  |
+----+-------+--------+------+
|  1 | user1 |      1 |   70 |
|  2 | user2 |      1 | NULL |
|  3 | user3 |      1 |   59 |
|  4 | user4 |      2 | NULL |
|  5 | user5 |      3 |   80 |
|  6 | user6 |      3 | NULL |
+----+-------+--------+------+
6 rows in set (0.00 sec)
```

1. 官方网址: www.yzmedu.com

2. 咨询电话: 400-105-8680

3. 官方微信: lampym