1. 不要用select \* 代替所有的select XX,XX字段，增加代码可读性
2. And优先级高于or，所以要or得先加小括号。否则会出现and xx再or的这种情况发生select detail\_id as id, product\_name as name, product\_price as price from order\_detail where product\_price>100 and (detail\_id=66 or detail\_id=77);
3. Like %A%，模糊查询，\_A,占位符
4. Order by 排序 desc，降序，asc，升序
5. 多个字段的Order by，优先级从左到右，用,隔开

select detail\_id as id, product\_name as name, product\_price as price from order\_detail order by product\_name,detail\_id;

1. Rand() 范围：0-1
2. Count（\*） 是包含了null值，count(字段)不计算null值
3. distinct 去重 select distinct product\_name,product\_price from order\_detail;
4. group by 分组 取得每个组最大值。select product\_name ,max(product\_price) as maxprice from order\_detail group by product\_name;
5. having 符合要求的部分用having函数，跟group by一起使用 select job,avg(sal) as maxsal from emp group by job having maxsal>2000;
6. SQL语句如下：select xxx from xxx where xxx group by xxx having xxx order by xxx
7. 笛卡尔积 若两张表连接查询没有任何条件限制，最终查到是两张表记录的乘积
8. 表的内(左，右)连接 select e.ename,d.dname from emp e inner(left,right) join dept d on e.deptno=d.deptno;
9. 内连接时，相应的有条件的会给连接上，但是如果在B表没找到条件时，A表的数据会丢失。左外连接则A表的数据都有，B表有的数据则显示，没有则显示null。
10. 自连接，select a.ename as yuangongename,b.ename as laobanename from emp as a left join emp as b on a.mgr=b.empno order by b.ename;
11. 全连接 多个表与A表相互关联，各自去连接即可 select a.ename,d.dname,b.ename,s.grade from emp a left join dept d on a.deptno=d.deptno left join emp b on a.mgr=b.empno left join salgrade s on a.sal between losal and hisal;
12. 子查询 把查询结果当成一个表 select t.avgsal,s.grade from (select e.deptno,avg(e.sal) as avgsal from emp e group by e.deptno) t left join salgrade s on t.avgsal between s.losal and s.hisal order by grade asc
13. Union 并集
14. Limit 取得前N个，以下是从第一行开始，取得前3行数据 select t.avgsal,s.grade from (select e.deptno,avg(e.sal) as avgsal from emp e group by e.deptno) t left join salgrade s on t.avgsal between s.losal and s.hisal limit 0,3;
15. VARCHAR和CHAR CHAR静态分配空间，VARCHAR动态分配空间
16. Create table xxx(); 创建表
17. Drop table xxx；删除表
18. create table emp\_copy as select \* from emp; 复制表
19. 插入数据 insert into t\_student(no,name,gender,birth,email) values(1,'zxw','1',str\_to\_date('2018-5-17','%%Y-%m-%d'),'zxw@qq.com');
20. 更新数据 update t\_student set name='w' where no=1;
21. 删除数据 delete from t\_student where no=1;
22. Alter table 修改表（新增字段，删除字段，修改字段数据长度，修改字段名，）alter table t\_student add cell int(16);alter table t\_student drop cell;alter table t\_student modify name varchar(16);alter table t\_student change name myname varchar(16);
23. 非空约束 not null
24. 唯一约束，可以列约束，也可以联合约束。 unique
25. 非空且唯一 not null unique
26. 主键 效果与not null unique，但多了索引功能。 Primary key
27. 主键自增 auto\_increment
28. 外键约束 在一对多的情况下，在多的表加上外键约束，减少数据冗余
29. On delete cascade 外键级联删除
30. On update cascade 外键级联更新
31. 存储引擎 MYSQL特有的！show engines; 查询数据库的存储引擎
32. Alter table xxx engine=存储引擎名称 来修改表的存储引擎
33. 创建索引 create (unique) index 索引名 on 表名（列名）
34. 创建视图 create view myview as select \* from t\_stu;
35. 视图作用 隐藏表的实现细节
36. 视图作用 提高检索的效率
37. 建表关系设计 多对多，用第三张表来对应关系；一对多，在多的表上加外键约束；一对一，分两张表存储，外键唯一
38. 开启事务 start transaction xxx xxx commit(rollback);