

1. 取出表里所有内容

a. `select * from emp`

2. 取出名字、月薪的内容

a. `select ename, sal from emp`

3. 取出名字、年薪

a. `select ename , sal * 12 from emp`

4. 年薪取别名

a. `select ename , sal * 12 annual_sal from emp`

b. `select ename , sal * 12 `annual sal` from emp`

c. `select ename , sal * 12 as `annual sal` from emp`

5. 计算总收入 (处理NULL值)

a. NULL是空值, 不是0

b. `select ename , sal * 12 + (case when comm is NULL then 0
else comm end) as annual_sal from emp`

6. distinct去掉重复值

a. `select distinct deptno from emp`

b. `select distinct deptno, job from emp`

7. where条件过滤

a. 取出deptno=10的记录 `select * from emp where deptno = 10`

b. 取出不等于10的名字和薪水 `select ename, sal from emp
where deptno <> 10`

c. 取出薪水位于800到1500之间的记录 `where sal between 800
and 1500`

d. 同上 `where sal >= 800 and sal <= 1500`

e. 在emp表中查询comm值为空值的数据 `where comm is null`

f. 不为空 `where comm is not null`

g. 薪水是800或者1500 `where sal in (800, 1500) ==> where
sal = 800 or sal = 1500`

h. 名字是SMITH或者KING `where name in ('SMITH', 'KING')`

i. 名字不是SMITH或者KING where name not in ('SMITH', 'KING')

j. 简单日期处理 where hiredate > '1981-02-20'

k. 模糊查询

i. 名字包含ALL where ename like '%ALL%'

ii. 名字第二个字母是L like '_L%'

iii. 名字中包含% ---- '%\%%'

8. 排序

a. select * from dept order by deptno

b. select * from dept order by deptno desc

c. select empno, ename from emp where deptno <> 10 order by empno asc; 当where混合使用时, 要先对数据进行过滤, 然后再进行升序或者降序的排序

d. select ename, sal, deptno from emp order by deptno asc; 在emp表中查询ename, sal, 和deptno的数据, 并且按照deptno的升序进行显示

e. select ename, sal, deptno from emp order by deptno asc, ename desc; 先按照deptno的升序进行一次排列, 然后对ename进行倒序排序再进行显示

9. 常用函数 <https://dev.mysql.com/doc/refman/5.7/en/func-op-summary-ref.html>

a. select lower(ename) from emp; 一个lower的函数, 接收了一个ename的实例, 作用是将字段中的字符串数据全部转换为小写 大写为upper()

b. select concat('www.', 'mashibing.', 'com') as url

c. select char_length('aaaa')

d. select substring('goodgoodstudy', 1, 5) //from下标从1开始

e. select ltrim/rtrim/trim(' aaa bbb ')

f. select abs(), ceil(), floor()

g. select round(3.14159, 3)

- h. `select truncate(3.14159, 3)`
- i. `select curdate()/curtime()/now()`
- j. `select month(hiredate) from emp`
- k. `select`
`monthname(d)/week(d)/year(d)/hour(time)/minute(time)`
- l. `select weekday(curdate()) / dayname(d) //`0 = Monday, 1 =
Tuesday, ... 6 = Sunday

10. 组函数(面试经常问)

- a. `select max(sal) from emp`
- b. `min(sal)`
- c. `avg(sal)`
- d. `sum(sal)`
- e. `count(*)`
- f. `count(comm)`

11. group by分组

- a. `select avg(sal) from emp group by deptno`; 按照deptno进行分组计算emp表中的sal平均值
- b. `select deptno, job, max(sal) from emp group by deptno, job`; 按照deptno和job两个字段进行分组
- c. 在所有记录中, 算出最高薪资的那个人
 - i. `select ename, max(sal) from emp`; 这条sql语句并不能求出所有人中薪资最高的那个人,
 - ii. 分析: 分组函数是多行输入, 但是只有一行输出, 这里的ename也会有很多人, 这时无法进行匹配
 - iii. `select ename from emp where sal = (select max(sal) from emp)`; 这里使用子查询, 解决了无法匹配的问题

- d. 求每个部门最高薪资 `select deptno, max(sal) from emp group by deptno;` 这条SQL语句可以求出deptno部门的最高薪资
- e. 求每个部门薪水薪水最高人的名字

i. `select deptno, max(sal) max_sal from emp group by deptno`

信息	结果 1	剖析	状态
deptno	max_sal		
10	5000		
20	3000		
30	2850		

ename	sal	deptno
SMITH	800	20
ALLEN	1600	30
WARD	1250	30
JONES	2975	20
MARTIN	1250	30
BLAKE	2850	30
CLARK	2450	10
SCOTT	3000	20
KING	5000	10
TURNER	1500	30
ADAMS	1100	20
JAMES	950	30
FORD	3000	20
MILLER	1300	10

ii. `select ename, e.deptno, sal from emp e join (select deptno, max(sal) max_sal from emp group by deptno) p where e.deptno = p.deptno and e.sal = p.max_sal`

1. 使用having对分组进行限制

- a. 按照部门编号进行分组，求分组之后每一个部门的平均薪水

i. `select avg(sal), deptno from emp group by deptno;`

- b. 这时，我们要求显示平均薪水大于2000的部门

i. select avg(sal), deptno from emp group by
deptno having avg(sal) > 2000;

2. 总结:

```
select xx from 表名;  
where xx xxxx; 对数据进行过滤  
group by;      分组  
having ;       对分组进行限制  
order by;      排序  
取数据, 过滤, 分组, 分组限制, 排序
```

要求薪水大于1200的雇员按照部门编号进行分组, 分组后的平均薪水大于1500, 查

询分组

内的平均工资, 按照工资的倒序进行排列

```
select avg(sal)  
from emp  
where sal > 1200  
group by deptno  
having avg(sal) > 1500  
order by avg(sal) desc;
```