1. 查找 employee中年龄大于等于 25并且 薪水大于3000的记录

select \* from employee where age>=25 and salary>3000;

1. 查找 employee 中 编号是偶数并且年龄大于20的记录

select \* from employee where id%2 =0 and age>20;

1. 更新age 字段的年龄都加9,要求编号在2-9 的记录，

update employee set age=age+9 where id between 2 and 9;

1. 查询员工号，姓名，工资以及工资提高百分20%后的结果

select id,name,salary,salary\*1.2 from employee;

1. 查询员工名中第三个字符为\_的员工名

select name from employee where name like '\_\_#\_%' escape '#';

1. 查询所有员工信息 并且 按先部门后编号降序排序

select \* from employee order by department\_id desc ,id desc

1. 选择工资不在5000-10000的员工的姓名和工资， 按工资降序

select name,salary from employee where salary not between 5000 and 1000

order by salary desc;

8.查询 各工种 job\_id 员工的工资的最大值 平均值 总和 并按 job\_id 降序排序

select job\_id, max(salary),avg(salary),sum(salary)

from employee

group by job\_id

order by job\_id desc;

9. 每个工种有奖金的员工的最高工资>5000的工种编号和最高工资，按最高工资升序

select job\_id,max(salary) as m

from employee

where bonus <>0

group by job\_id

having m>5000

order by m asc;

10. 查询哪个部门的员工人数大于2

select department\_id,count(\*) as c

from employee

group by department\_id

having c>=2;

11. 查询平均成绩大于60分的同学的学号和平均成绩

SELECT sid,AVG(score) as a

FROM sC

GROUP BY sid

HAVING a>=60;

12．查询所有同学的学号、姓名、选课数、总成绩

SELECT s.sid AS "学号", s.sname AS "姓名", COUNT(sc.cid) AS "课程数目", SUM(sc.score) AS "总分数"

FROM student as s

INNER JOIN sC as sc

ON s.sid=sc.sid

GROUP BY s.sid

order by 总分数 desc;

13. 查询哪个部门的员工大于2的部门名和员工个数

select department\_name,count(\*) as c

from employee as e

inner join department as d

on e.department\_id = d.department\_id

group by department\_name

having c>2;

14. 查询学过“张三”老师课的同学的学号、姓名

SELECT s.sid AS "学号", s.sname AS "姓名"

FROM student s

INNER JOIN sC as sc ON s.sid=sc.sid

INNER JOIN course as c ON c.cid=sc.cid

INNER JOIN teacher as t ON t.tid=c.tid

WHERE t.tname="张三";

15. 查询各科成绩最高和最低的分：以如下形式显示：课程ID，最高分，最低分

SELECT cid AS "课程ID", MAX(score) AS "最高分", MIN(score) AS "最低分"

FROM sC as sc

GROUP BY cid

16. 查询员工名、部门名，工种名并按部门名降序

select e.name,d.department\_name,j.job\_name

from employee as e

inner join department as d on e.department\_id = d.department\_id

inner join job as j on e.job\_id = j.job\_id

order by e.department desc;

19. 返回公司工资最少的员工的name,job\_id和salary

select name,job\_id,salary from employee where salary =(

select min(salary) from employee

);

20. 返回job\_id与7号员工相同，salary比8号员工多的员工 姓名，job\_id 和工资

select name,job\_id,salary from employee where job\_id=(

select job\_id from employee where id=1

) and salary >(

select salary from employee where id=2

);

20. 查询最低工资大于0002号部门最低工资的部门编号和其最低工资

select department\_id,min(salary) as m

from employee

group by department\_id

having m>(

select min(salary) from employee where department\_id=0002

);

21查询(每个班第一名同学)的名字和分数

名字 班级 分数

A 1 88

B 1 90

C 1 92

D 2 93

E 2 85

一列多行(列子查询)

select name,score from info where score in (select max(score) from info group by class);

22 查询（每个部门的最低工资）的员工的名称,部门编号和薪水

select name,department\_id,salary from employee

where salary in(select min(salary) from employee group by department\_id);

1. 单选题
   1. 将数值123456赋值给float(5,3);的结果是（ C ）；

A. 12345.600 B. 1.23456

C. 错误 D. 1.2346e5

* 1. 如果将456789赋值给tinyint变量将引发的错误是（ D ）；

A. ERR 1062 B. ERR 1048

C. ERR 10062 D. ERR 1264

* 1. 下列描述char和varchar正确的是（ B ）；

A. Char必须按长度赋值

B. varchar的实际存储长度与赋值字符长度有关，但定义长度是最大长度；

C. Char可以保存超过定义长度的字符

D. Varchar可以保存超过定义长度的字符

* 1. 在Mysql中修改表结构的命令是（ D ）；

A. Modify table B. modify column

C. Drop table D. alter table

* 1. 在标准SQL分类中（ B ）不属于DML；

A. UPDATE B. SELECT

C. DELETE D. INSERT

* 1. 关系型数据库中的关系就是（ B ）

1. 表之间的关联 B. 二维表
2. 数据库之间的关联 D. 数据库和表的联系
   1. 以下描述正确的是（ B ）

A. 在windows下MySql服务必须经过安装程序才能使用

B. 在windows下MySql服务可以通过安装和解压两种方式运行；

C. MySql只能在Windows环境下执行

D. MySql在Linux下能用其它软件替代

* 1. MySql安装后，默认占用端口是（ A ）

A. 3306 B. 8066；

C. 8086 D. 1503

* 1. MySql安装时我们为了更好的处理中文，一般选用的编码是（ B ）

A. Iso-8859-1 B. utf8；

C. Latin1 D. Unicode

* 1. 有符号tinyint的最大值是（ C ）；

A. 128 B. 256

C. 127 D. 255

* 1. 语句select 2.3e5;的显示结果是（ B ）；

A. 2.300000 B. 230000

C. 错误 D. 11.5

* 1. 执行insert命令时，如果未指定列名则表示（ C ）；

A. 可以随便给定值 B. 只需给定主键值

C. 必须给定每个字段的值 D. 只需给定有约束限制的列提供值

* 1. 执行insert，update，delete命令后的返回值都是一个（ A ），代表更新的结果；

A. 整数 B. 结果集

C. 毫秒值 D. 二维表

* 1. 执行一个只带有order by子句的delete命令，表示（ C ）；

A. 删除排序的第一条 B. 删除排序的最后一条

C. 删除所有 D. 不删除

* 1. 删除成绩（grade）最低的学生记录，使用命令（ D ）；

A. Delete from student order by grade

B. Delete from student order by grade desc

C. Delete from student order by grade desc limit 1

D. Delete from student order by grade limit 1

* 1. 下列有关索引描述正确的是（ D ）；

A. 各种索引中，只有主键具有唯一性要求；

B. 主键列允许有null值；

C. 主键索引仅可以提高主键列的检索速度；

D. Unique 索引列允许有null值；

* 1. 当给字段unit varchar(3) 赋值‘工业大学’时会引发异常（D ）；

A. 1062 B. 1264

C. 3306 D. 1406

* 1. 存储过程中如果需要给调用者返回数据，需要使用（ B ）完成；

A. Return值 B. out 类型的参数

C. Declare 变量 D. in 类型的参数

* 1. 以下聚合函数求数据总和的是( B )

A．MAX

B．SUM

C．COUNT

D．AVG

* 1. SELECT语句的完整语法较复杂，但至少包括的部分是( B )

A．仅SELECT

B．SELECT，FROM

C．SELECT，GROUP

D．SELECT，WHERE

* 1. 修改数据库表结构用以下哪一项( D )

A．UPDATE

B．CREATE

C．UPDATED

D．ALTER

* 1. 删除用户账号命令是( A )

A．DROP USER

B．DROP TABLE USER

C．DELETE USER

D．DELETE FROM USER

* 1. SQL是一种( C )语言。

A． 函数型

B．高级算法

C．关系数据库

D． 人工智能

* 1. 若要在基本表S中增加一列CN（课程名），可用( B )

A．ADD TABLE S ALTER（CN CHAR（８））

B．ALTER TABLE S ADD（CN CHAR（８））

C．ADD TABLE S（CN CHAR（８））

D．ALTER TABLES S （ADD CN CHAR（８））

* 1. 按照姓名降序排列( B )

A．ORDER BY DESC NAME

B．ORDER BY NAME DESC

C．ORDER BY NAME ASC

D．ORDER BY ASC NAME

* 1. 返回当前日期的函数是( D )

A．curtime()

B．adddate()

C．curnow()

D．curdate()

* 1. SELECT COUNT(SAL) FROM EMP GROUP BY DEPTNO;意思是( D )

A．求每个部门中的工资

B．求每个部门中工资的大小

C．求每个部门中工资的综合

D．求每个部门中工资的个数

* 1. 条件“BETWEEN 20 AND 30”表示年龄在20到30之间，且( D )

A．包括20岁不包括30岁

B．不包括20岁包括30岁

C．不包括20岁和30岁

D．包括20岁和30岁

* 1. 从GROUP BY分组的结果集中再次用条件表达式进行筛选的子句是( C )

A．FROM

B．ORDER BY

C．HAVING

D．WHERE

* 1. 为数据表创建索引的目的是( A )

A．提高查询的检索性能

B．归类

C．创建唯一索引

D．创建主键

* 1. MySQL中，预设的、拥有最高权限超级用户的用户名为( D )

A．test

B．Administrator

C．DA

D．root

* 1. 以下语句不正确的是( C )

A．select \* from emp;

B．select ename,hiredate,sal from emp;

C．select \* from emp order deptno;

D．select \* from emp where deptno=1 and sal<300;

* 1. 格式化日期的函数是( B )

A．DATEDIFF()

B．DATE\_FORMAT()

C．DAY()

D．CURDATE()

* 1. 一张表的主键个数为( C )

A．至多3个

B．没有限制

C．至多1个

D．至多2个

* 1. 以下表示可变长度字符串的数据类型是( C )

A．TEXT

B．CHAR

C．VARCHAR

D．EMUM

* 1. 拼接字段的函数是( D )

A．SUBSTRING()

B．TRIM()

C．SUM（）

D．CONCAT()

* 1. 下列说法错误的是( D )

A．GROUP BY 子句用来分组

B．WHERE 子句用来筛选 FROM 子句中指定的操作所产生的行。

C．聚合函数一般和group by 一起使用

D．HAVING 子句用来从FROM的结果中筛选行

* 1. 20. 例如数据库中有A表，包括学生，学科，成绩三个字段 , 数据库结构为

学生 学科 成绩

张三 语文 60

张三 数学 100

李四 语文 70

李四 数学 80

李四 英语 80

如何统计最高分>80的学科( C )

A．SELECT MAX(成绩) FROM A GROUP BY学科 HAVING MAX(成绩)>80;

B．SELECT学科 FROM A GROUP BY学科 HAVING成绩>80;

C．SELECT学科 FROM A GROUP BY学科 HAVING MAX(成绩)>80;

D．SELECT学科 FROM A GROUP BY学科 WHERE MAX(成绩)>80;

* 1. 创建数据库命令的基本格式（ create ） ；
  2. 在MySql5.7中数据库引擎是（ INNODB ）
  3. 语句select year(20191122);的显示结果是（ 2019 ）；
  4. 当一个表中定义了timestamp字段，那么在执行（ insert ） 和 （ update ）命令时自动记录操作的系统时间；
  5. 在Mysql中列级约束包括主键约束，（ 非空约束 ），默认值约束；
  6. 使用delete删除所有记录以后，自动递增列将按照（ 删除前的最大值 ）开始计数；
  7. 事物的性质包括 原子性、（ 一致性 ）、（ 持久性 ）、（ 隔离性 ）。
  8. SELECT 语句中，使用（ where ）子句可提取满足条件的记录。
  9. 查询中需要统计记录个数时，应使用（ count(\*) ）聚合函数。
  10. 使用（ call ）命令调用存储过程。
  11. 数据库视图被称为“虚拟表”，创建视图格式是（ create view ）视图名 AS SELECT语句。
  12. 命令alter table student（ add ）unique(tel) ;实现给tel列添加唯一性约束；
  13. 在标准SQL分类中CREATE,（ ALTER）,（ DROP ）属于DDL；
  14. 在关系数据库中的标准语言叫做“结构化查询语言”，简称为（ SQL ）；
  15. 在使用排序短语order by进行查询的默认顺序是（ 升序 ）；
  16. 在数据库中，直接备份数据库的数据文件称为（ 物理备份 ），导出数据库对象的结构和数据称为（ 逻辑备份）；
  17. 创建数据库基本的命令是（ create database ）数据库名。
  18. 在SQL中使用（ create table ） 命令建立基本表结构。
  19. 组合索引遵循（ 左 ）优先原则,因为索引方法是Btree树状的，搜索时需要从根节点出发，上层节点对应靠左的值
  20. 在标准SQL分类中（ grant ）,revoke属于DCL。

1. 简单题：
   1. 在如下表中完成指定的操作：

* CREATE TABLE `book` (
* `bid` int(4) NOT NULL,
* `bookName` varchar(30) DEFAULT NULL,
* `author` varchar(20) DEFAULT NULL,
* `price` decimal(5,2) DEFAULT NULL,
* PRIMARY KEY (`bid`)
* )
  + 给主键列bid添加自动递增；
  + 自动递增的初值设定为20210001
* 答案：
  + alter table book modify COLUMN bid TINYINT auto\_increment;
  + alter table book auto\_increment=20210001;
  1. **在MySql中使用语句完成复制表的操作：复制student到s1表；**
* 答案：
  + **create table s1 like student;**
  + **insert into s1 SELECT \* from student;**
  1. 使用MySql进行分页查询，假设每页4行，写出前两页的查询语句；
* 答案：
  + 第一页：select \* from book order by price limit 0,4;
  + 第二页：select \* from book order by price limit 4,4;
  1. 使用mysqldump进行逻辑备份，书写命令完成备份和恢复所有数据库的操作（备份文件all.sql放置在D盘根目录，用户root，密码127）；
* 答案：
  + mysqldump -uroot -p127 **-A** >d:/all.sql
    - 或者:mysqldump -uroot -p127 --all-databases >d:/all.sql
  + mysql -uroot -p127 <d:/all.sql
  1. 定义触发器，实现检查插入学生表student的成绩grade列，要求grade值不超过100；
* 答案：

delimiter $$

create TRIGGER stugrade

BEFORE INSERT

on student for EACH ROW

BEGIN

DECLARE msg varchar(100);

if NEW.grade >100 THEN

set msg=concat('输入成绩：',NEW.grade,' 超出范围！');

SIGNAL SQLSTATE 'HY000' SET MESSAGE\_TEXT=msg;

end if;

end $$

delimiter ;

* 1. 创建 employee 的表结构

要求：

* 1. id int 编号(主键，自动增长)
  2. job\_id int 工种
  3. name varchar(30) 员工名称
  4. department\_id int 部门编号(001,002)
  5. salary decimal(10,2) 薪水
  6. bonus decimal(10,2) 奖金
* 答案：

create table employee(

id int unsigned primary key auto\_increment,

job\_id int not null,

name varchar(30) not null unique,

department\_id int not null,

salary decimal(10,2) not null default 1000,

bonus decimal(10,2) not null default 0

);

* 1. 写出给表employee结构添加两条数据

要求：

* 1. job\_id 值 2, name 值张三，department\_id 值1,salary 值5000，bonus 值 800；
  2. job\_id 值 3, name 值李四，department\_id 值1,salary 值4000，bonus 值 100；
* 答案：

insert employee (job\_id,name,department\_id,salary,bonus) values(2,’张三’,1,5000,800);

insert employee (job\_id,name,department\_id,salary,bonus) values(3,’李四’,1,4000,1000);

* 1. 对表employee查询语句；

要求：

1. 查找 薪水(salary)大于等于 4500并且 奖金(bonus)大于 500的记录
2. 查询部门编号(department\_id)为1 的员工个数

* 答案：

select \* from employee where salary>=4500 and bonus>500;

select count(\*) from employee where department\_id=1;

* 1. 对表employee分页查询，假设每页2 记录，写出前两页的查询语句；
* 答案：

select \* from employee limit 0,2;

select \* from employee limit 2,2;

* 1. 对表employee查询分组和排序

要求：

1. 查询所有部门的编号，员工数量和工资的平均 并按平均工资降序

* 答案：

select department\_id,count(\*),avg(salary) as a

from employee

group by department\_id

order by a desc;

1. 综合题：

根据下图一对多，多对多表数据模型结构，完成SQL语句操作

* 1. 使用存储过程，实现批量插入，根据次数插入到cate分类表中多条记录
* 答案：

delimiter $

CREATE PROCEDURE pro\_while1(IN insertCount INT)

BEGIN

DECLARE i INT DEFAULT 1;

WHILE i<=insertCount DO

INSERT INTO cate(cname) VALUES(CONCAT('商品分类',i));

SET i=i+1;

END WHILE;

END $

delimiter ;

CALL pro\_while1(100);

* 1. 对cate,goods 完成多表联合查询语句及where子查询

要求：

1. 查询 商品标题（goods中），商品分类名称 (cate中）。

select g.title,c.cname

from goods as g

inner join cate as c on g.cate\_id = c.id;

1. 哪些商品的价格比商品1 高(goods表)？

* 答案：

select \* from goods where price >(

select price from goods where name='商品1');