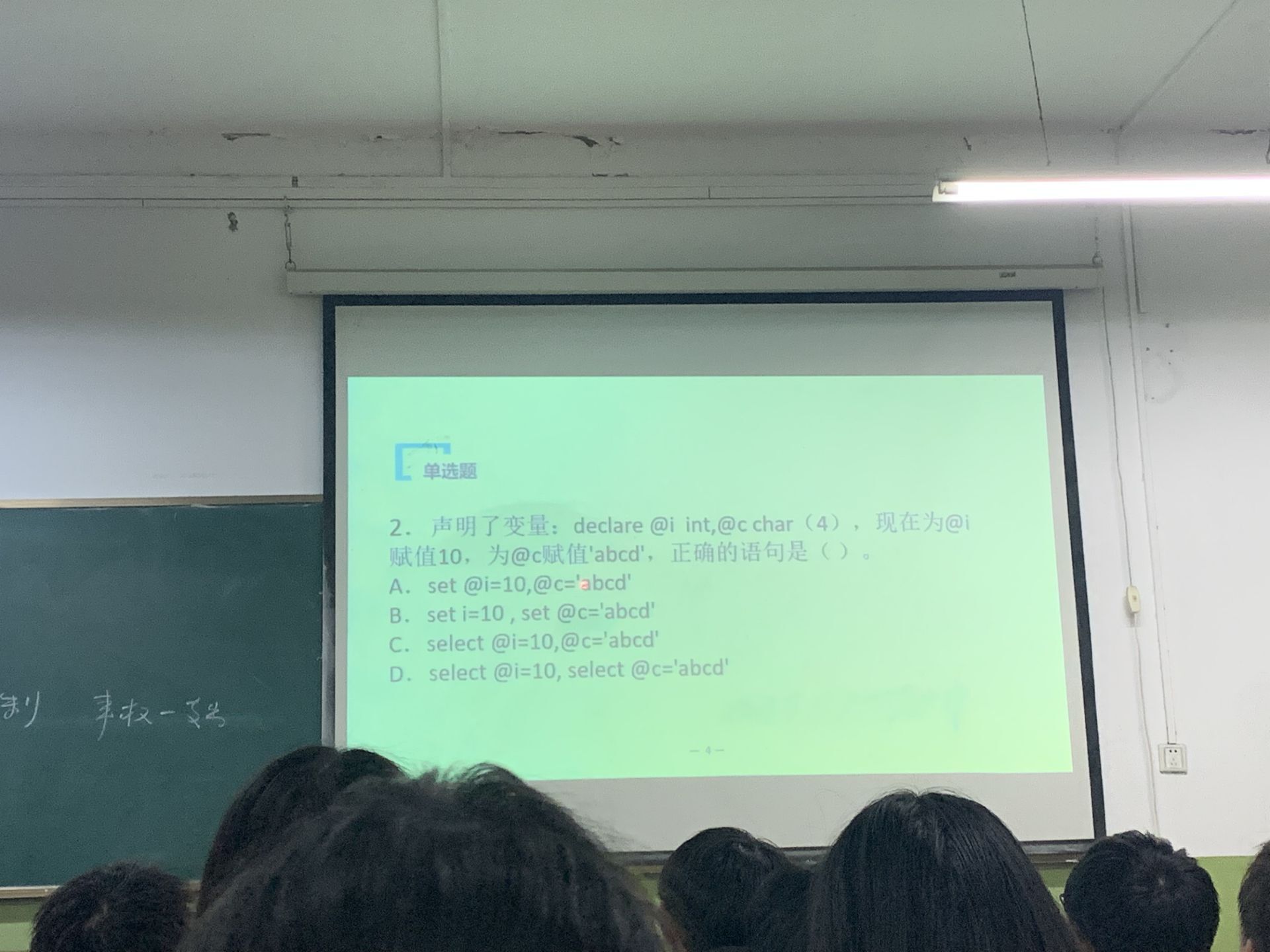
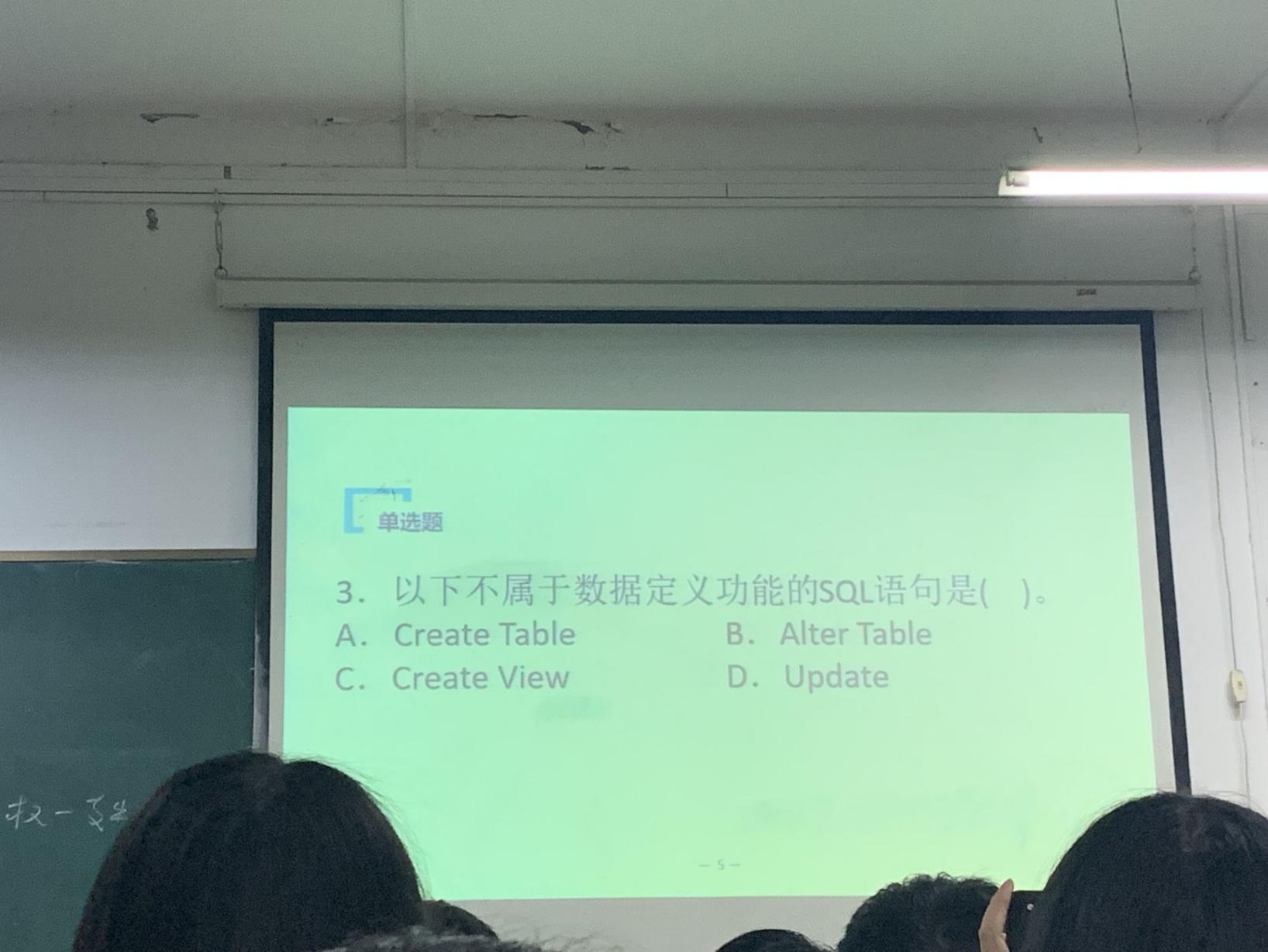
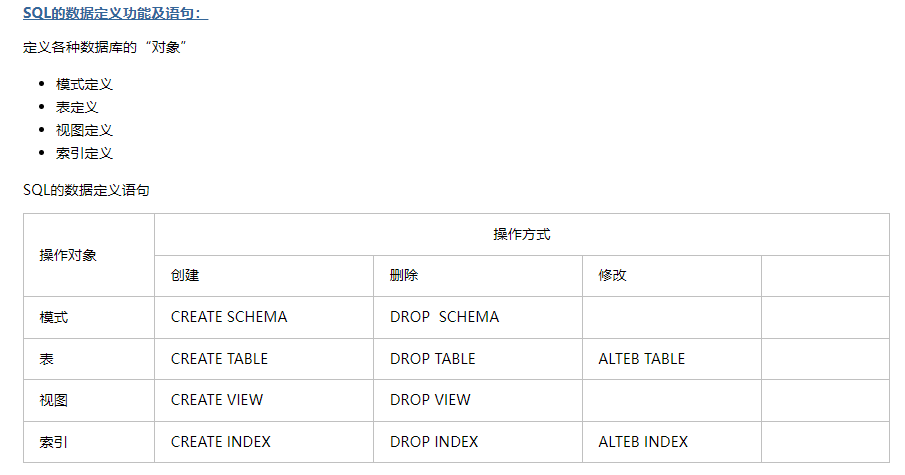
1. 对于局部变量，text，ntext，image数据类型无效。选择tabal
2. C

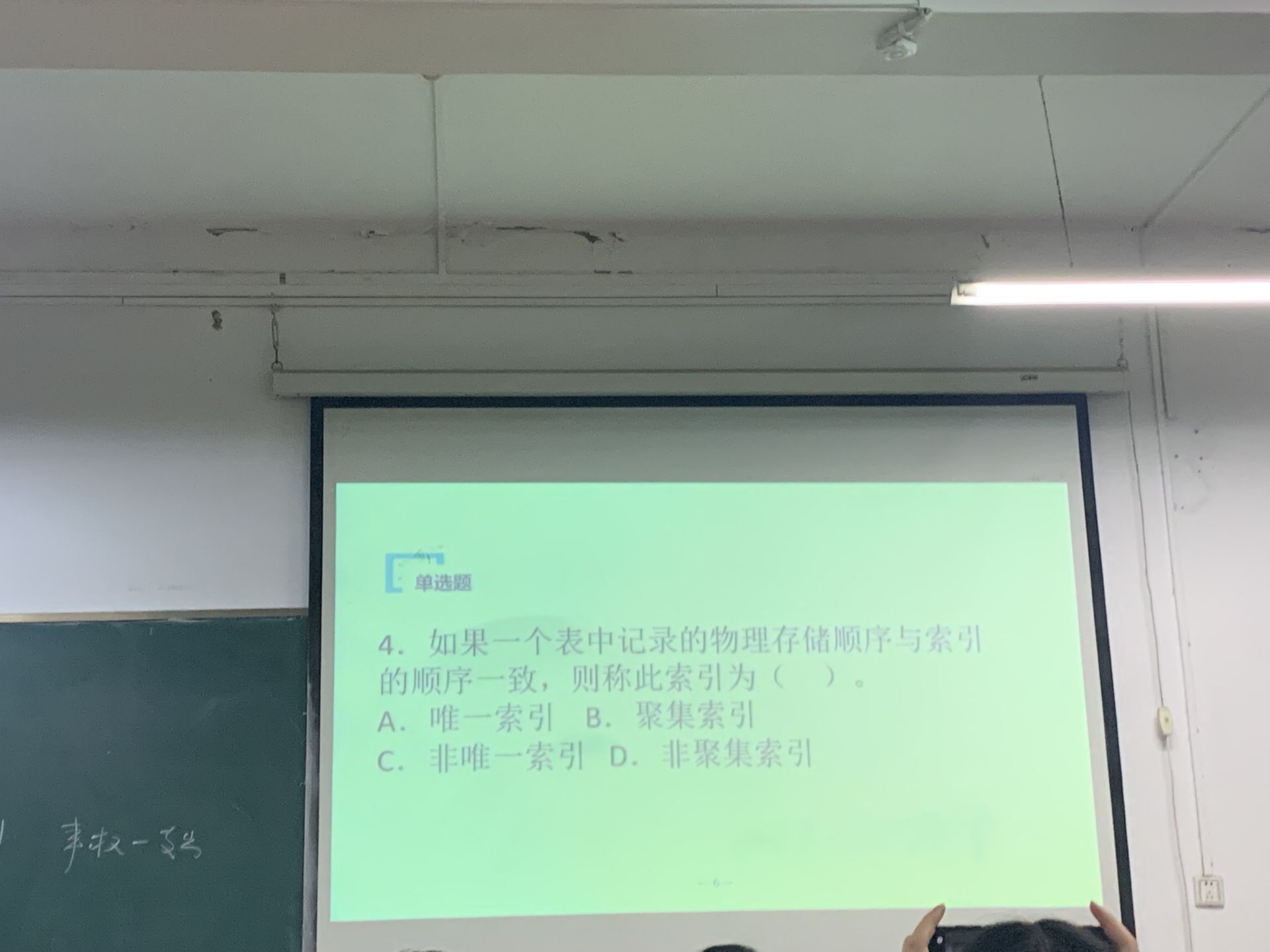


1. D

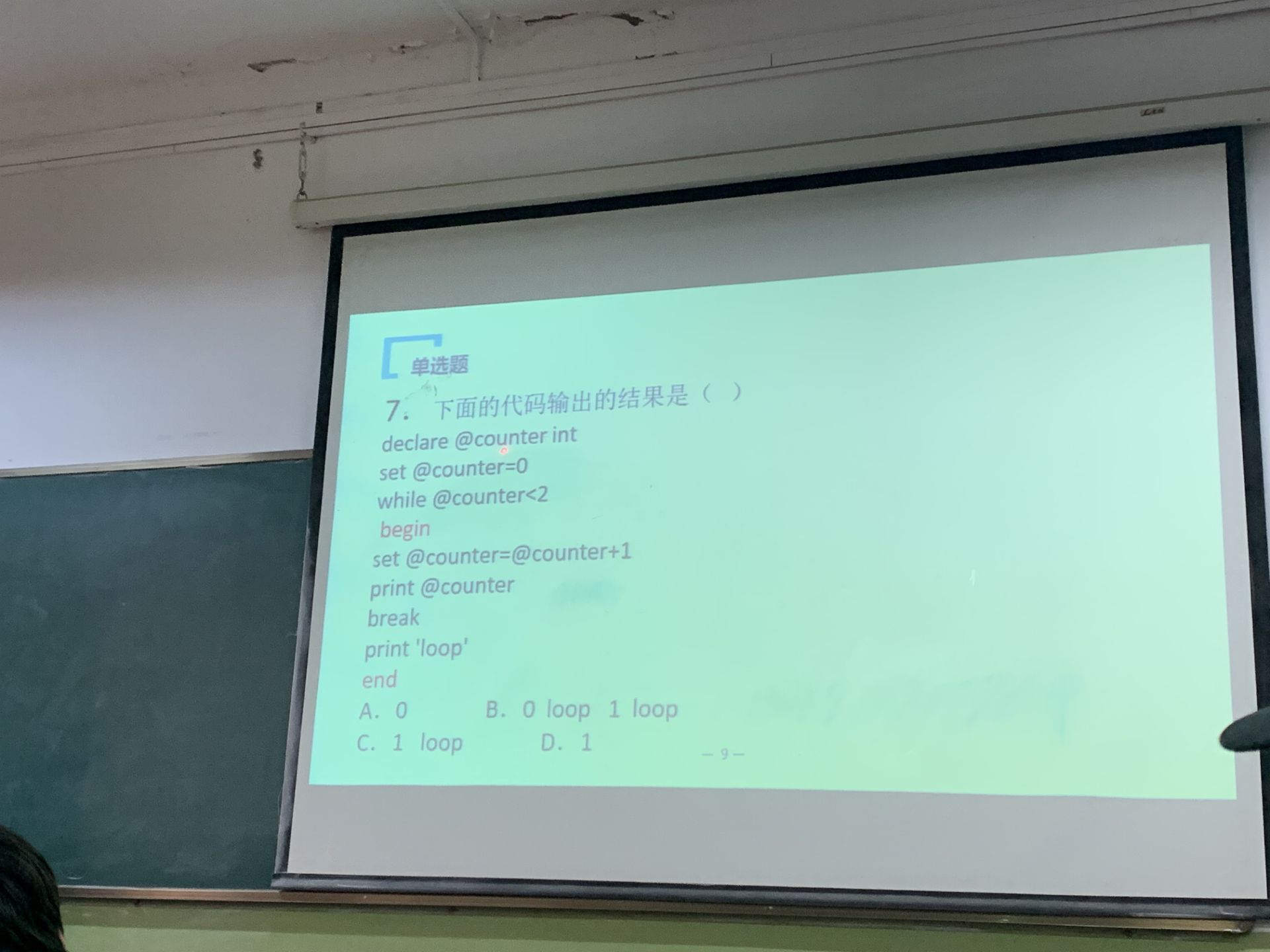




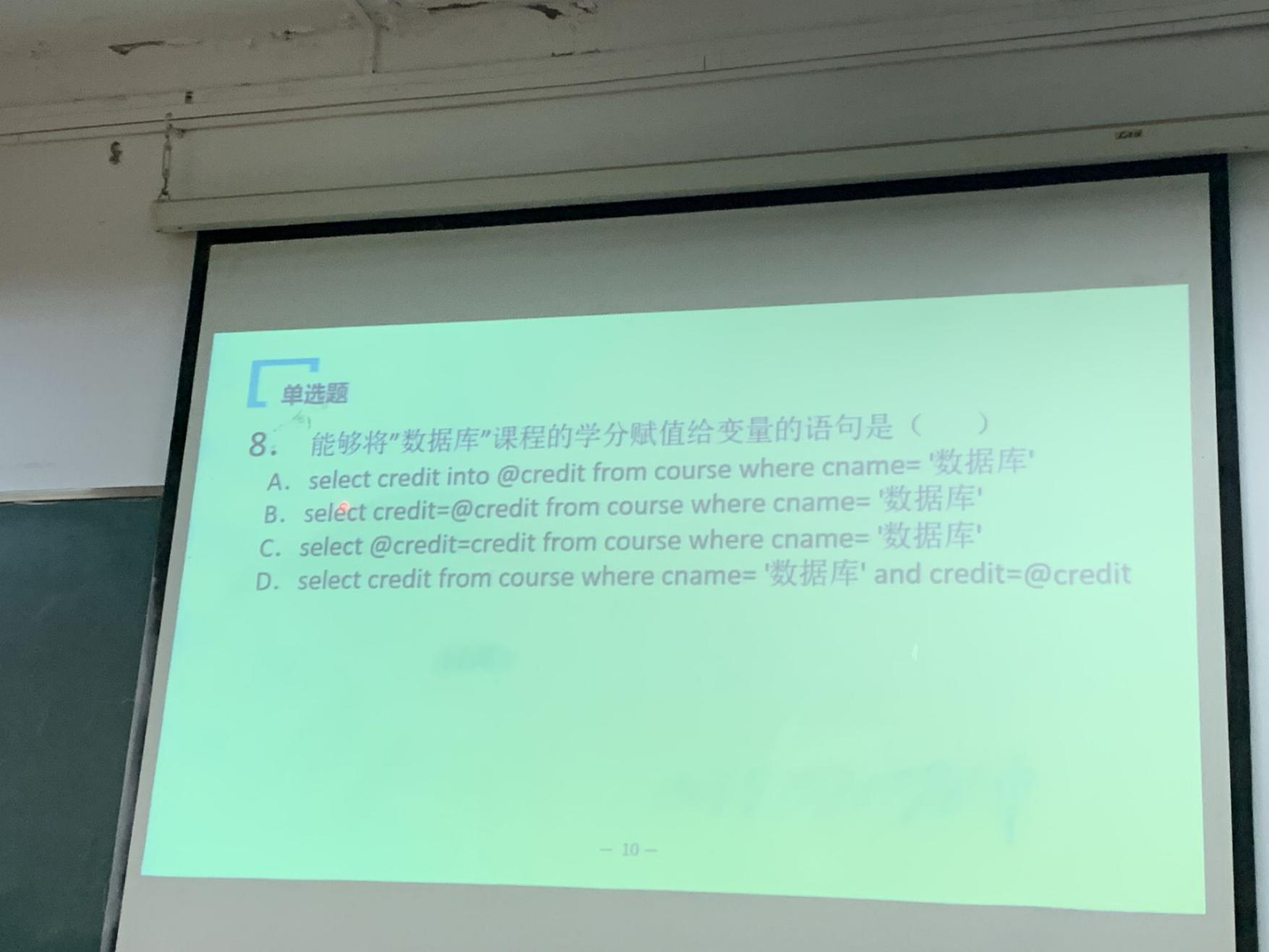
4.B



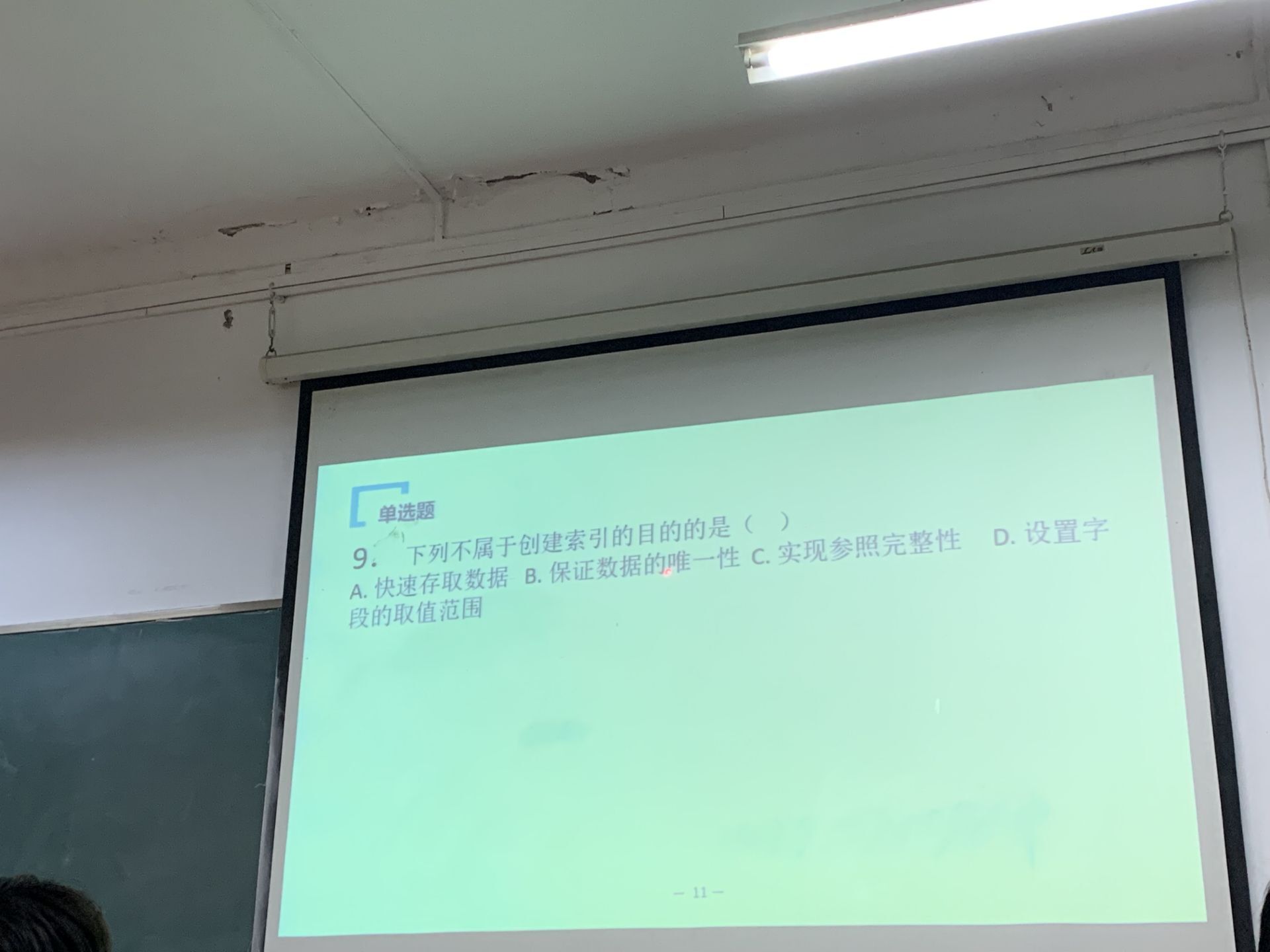
7. D



8.C



9.D

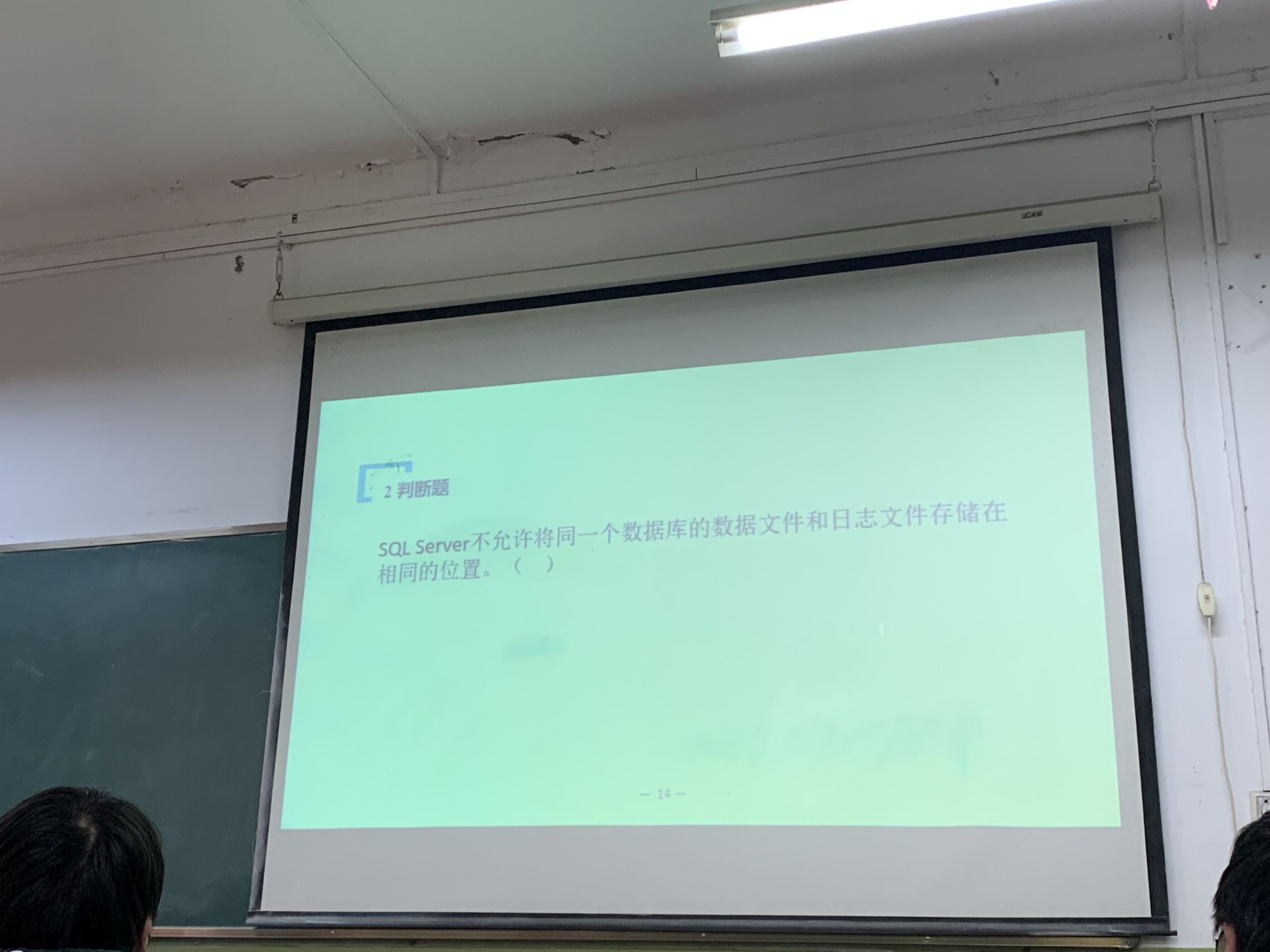


10.C

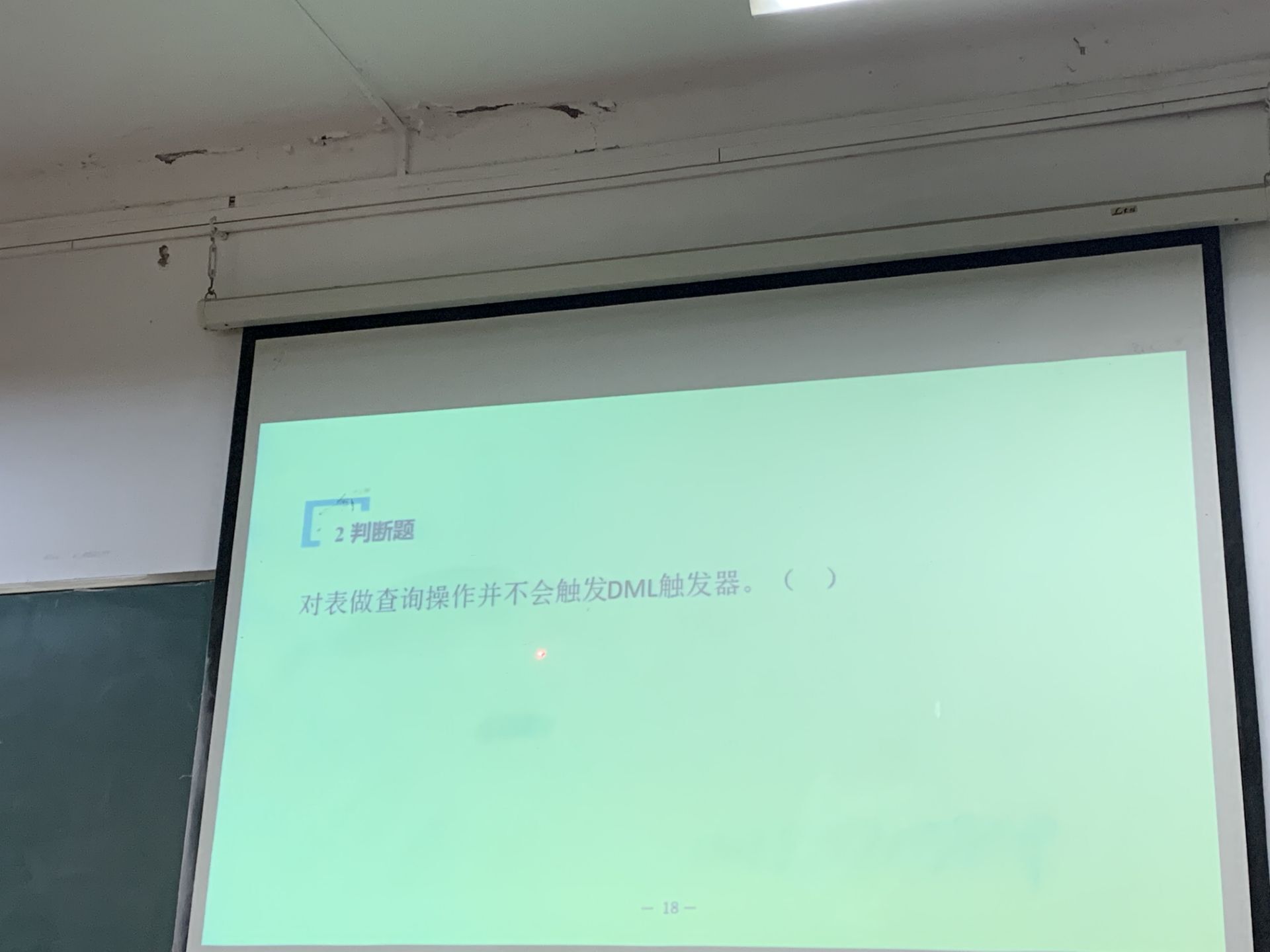


判断：

False

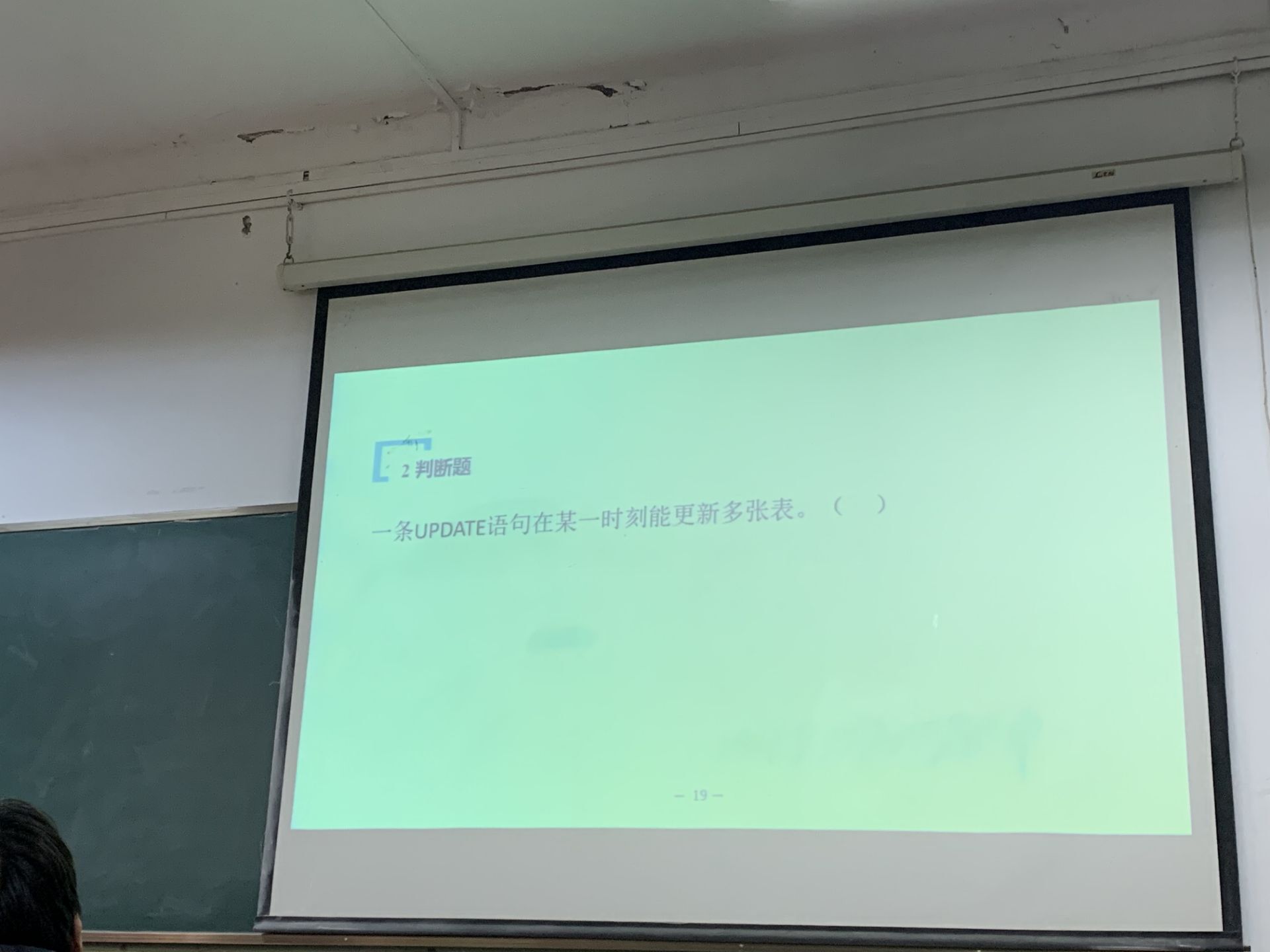


TRUE

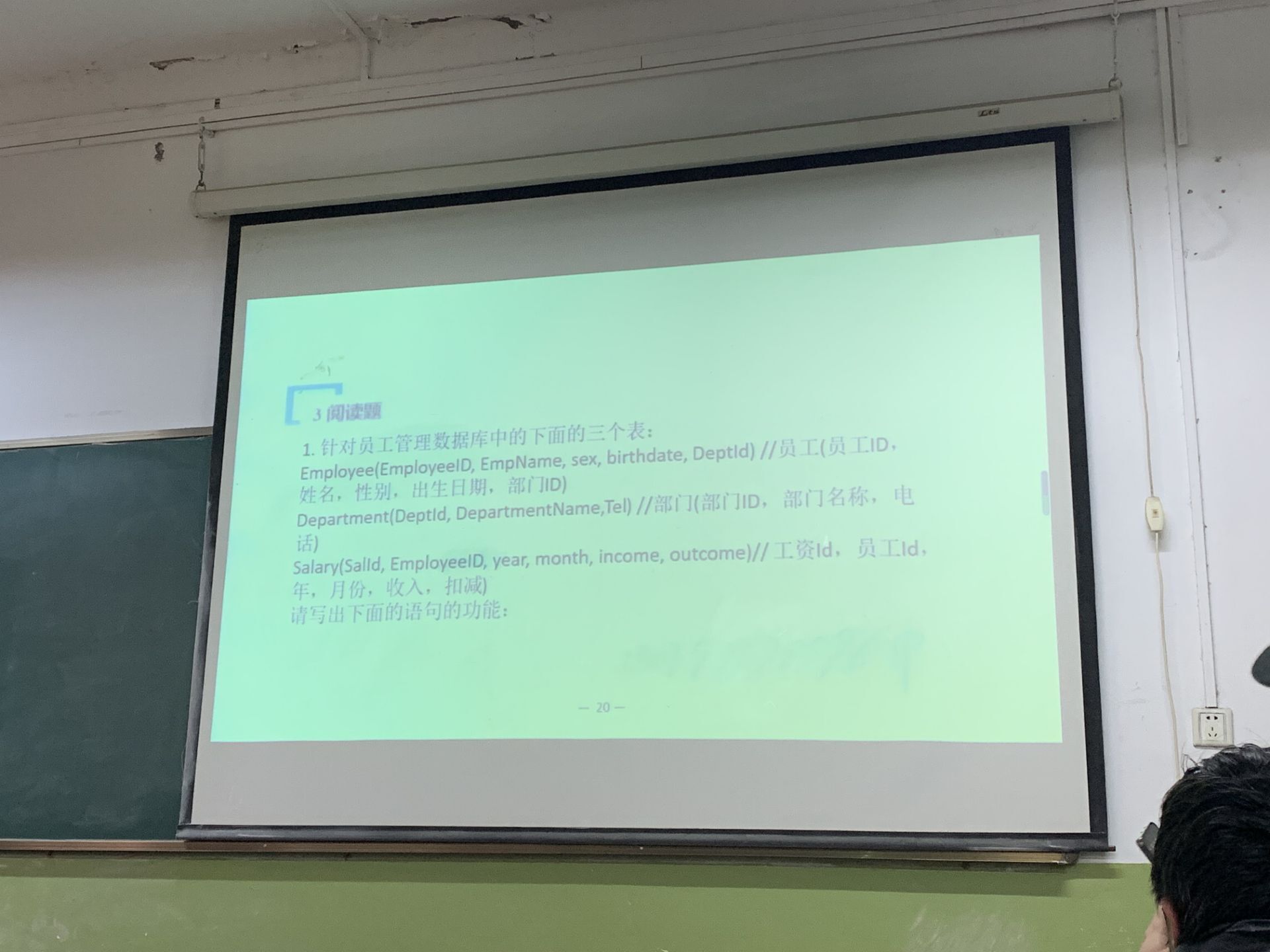


**DML触发器就是普通的 INSERT / UPDATE / DELETE 触发器**

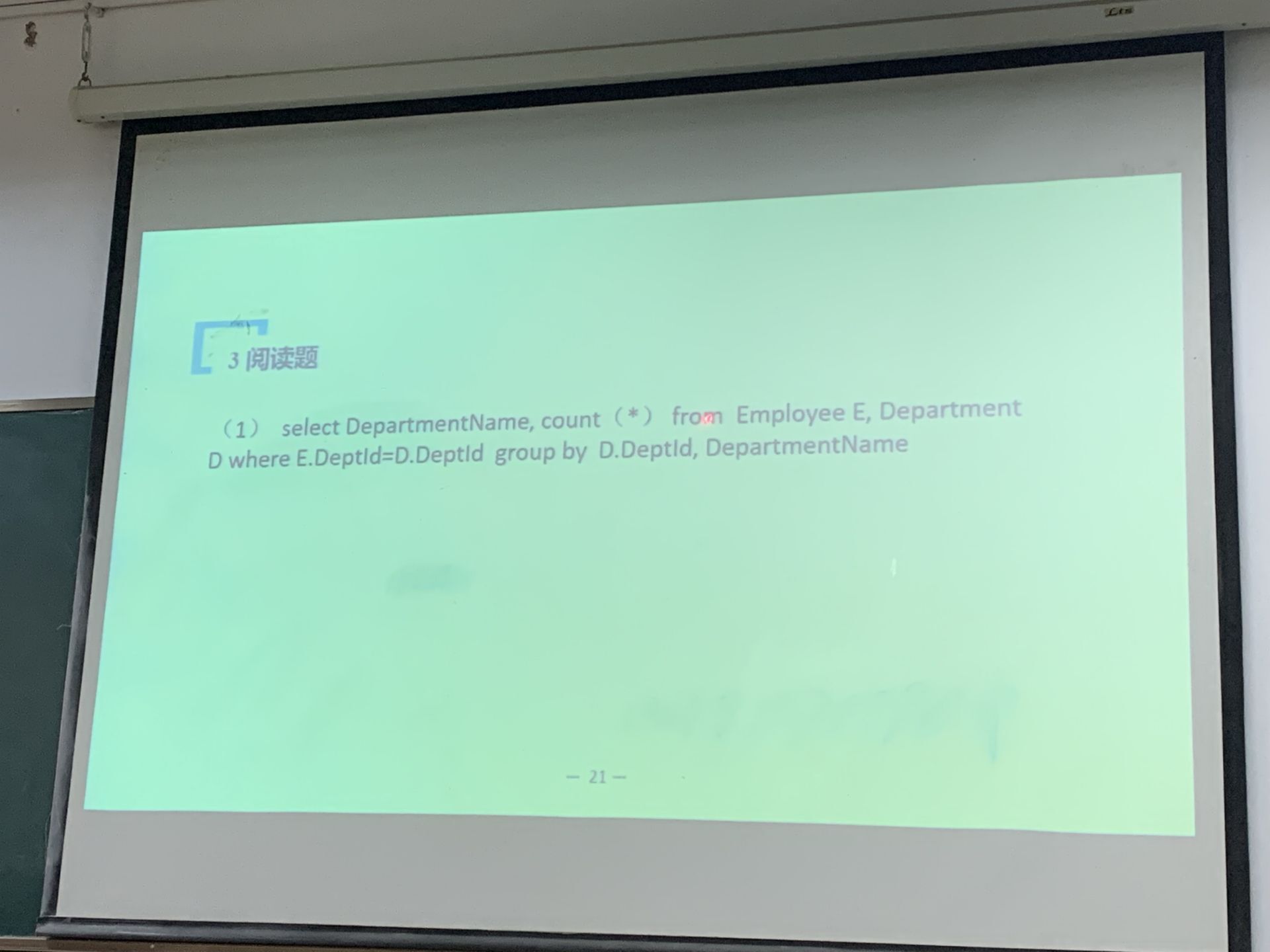
FALSE



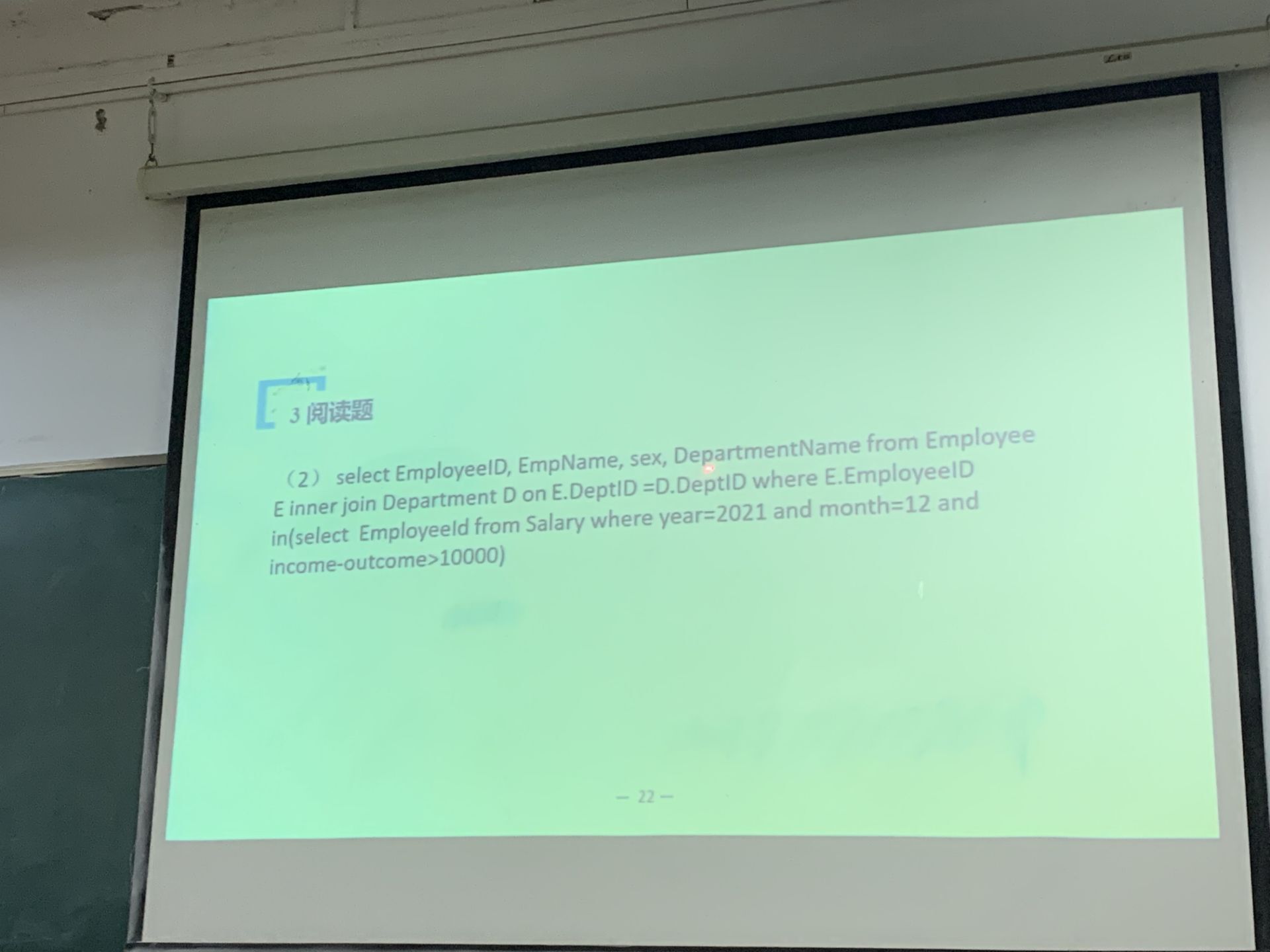
阅读题：



1. 查询各部门的人数，按照部门ID和部门名称进行分组，展示部门名称和部门人数字段



2. 从员工表和部门表中查询2021年12月份实际收入（收入-支出）大于10000的员工ID，姓名，性别和部门名称

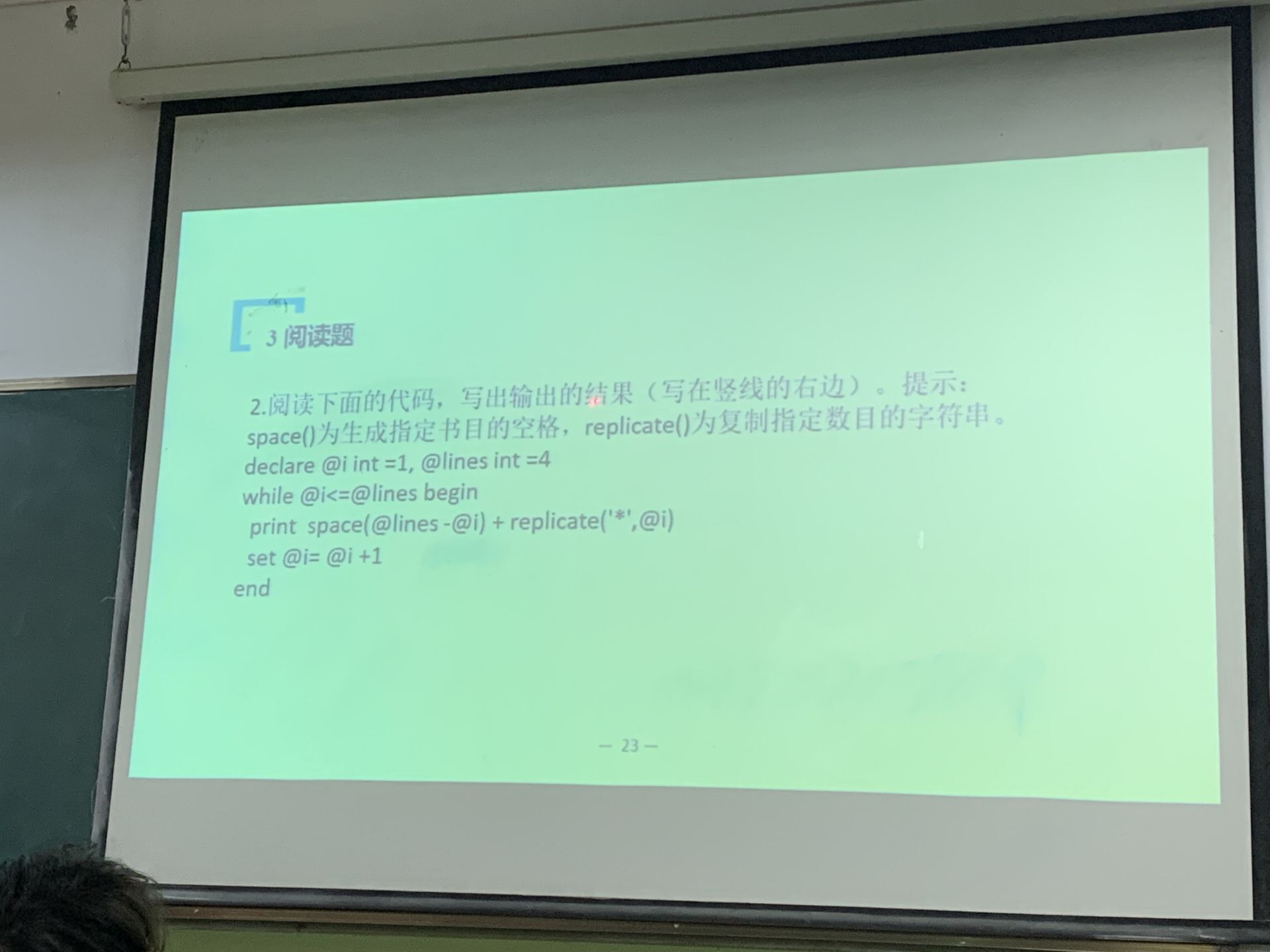


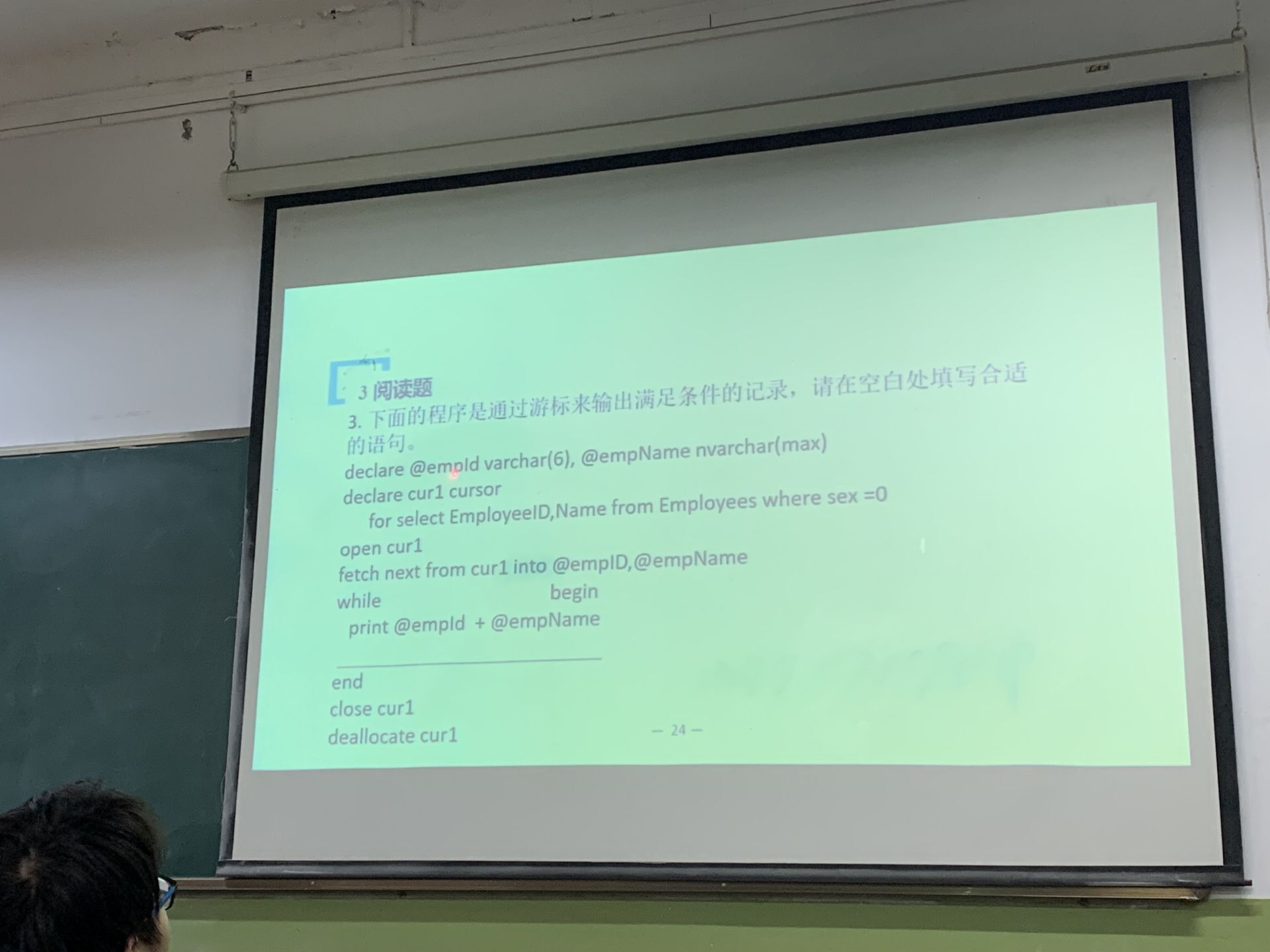
▢▢▢\*

▢▢\*\*

▢\*\*\*

\*\*\*\*

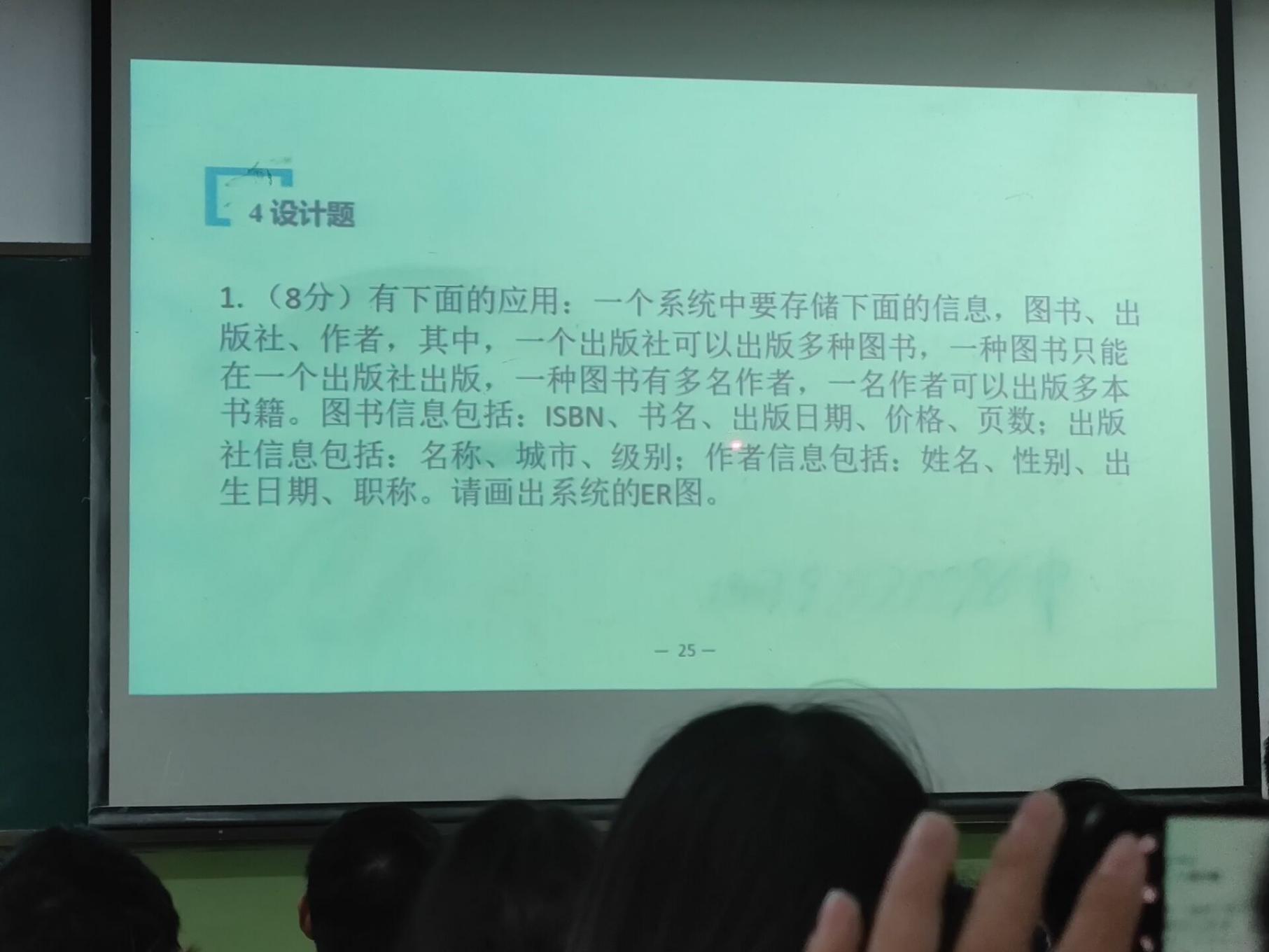


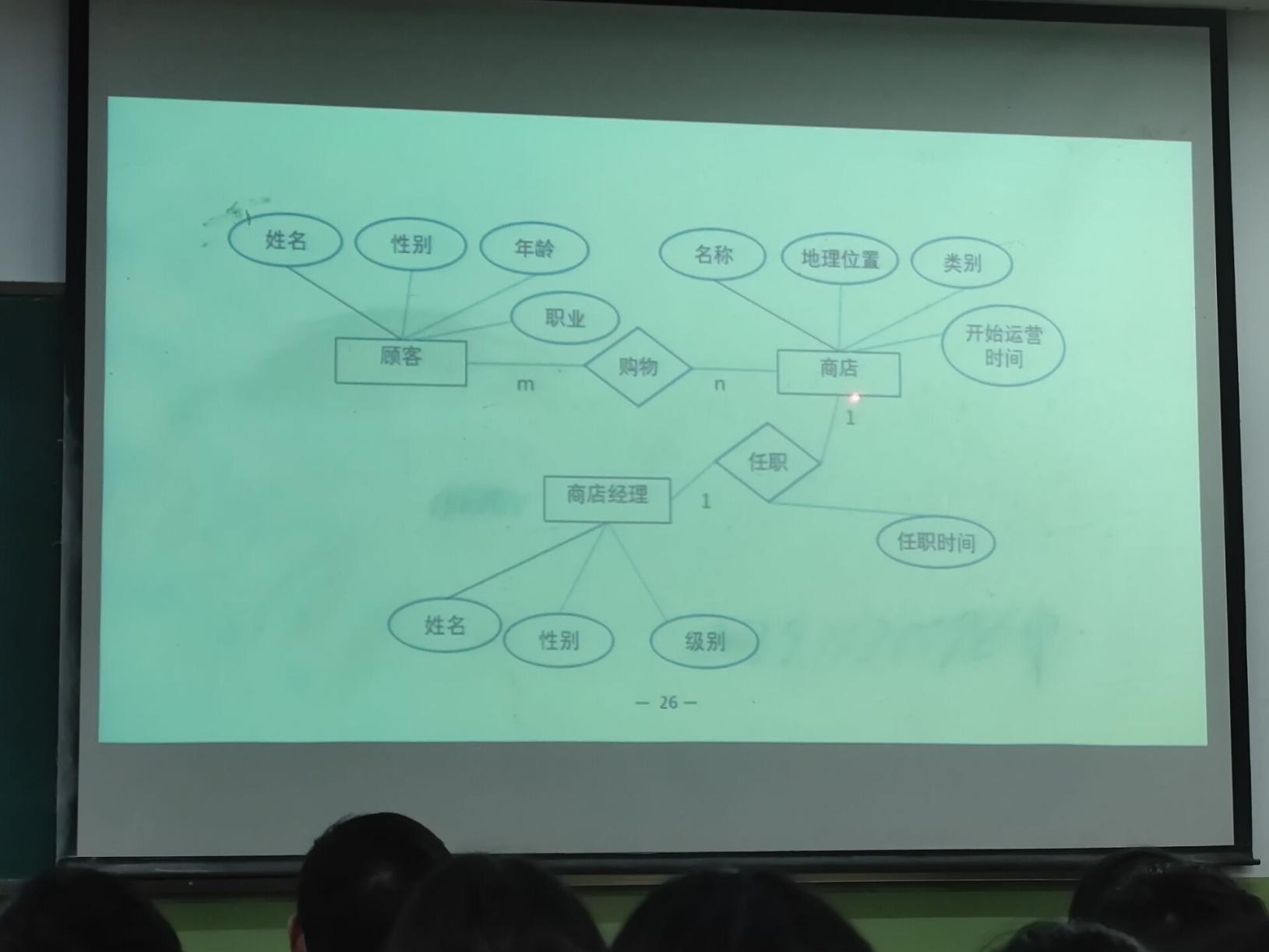


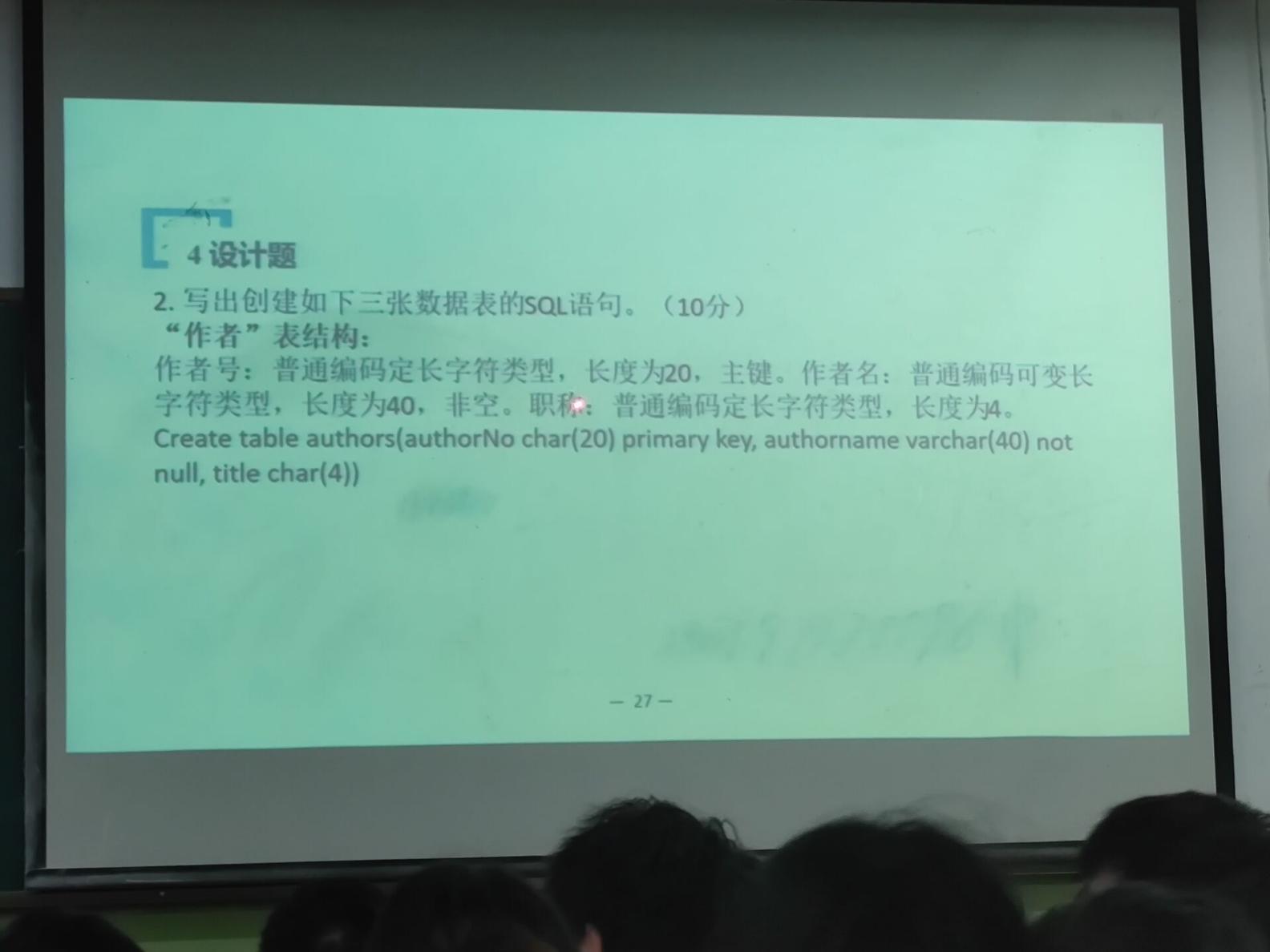
@@FETCH\_STATUS=0

fetch next from cur1 into @empID,@empName

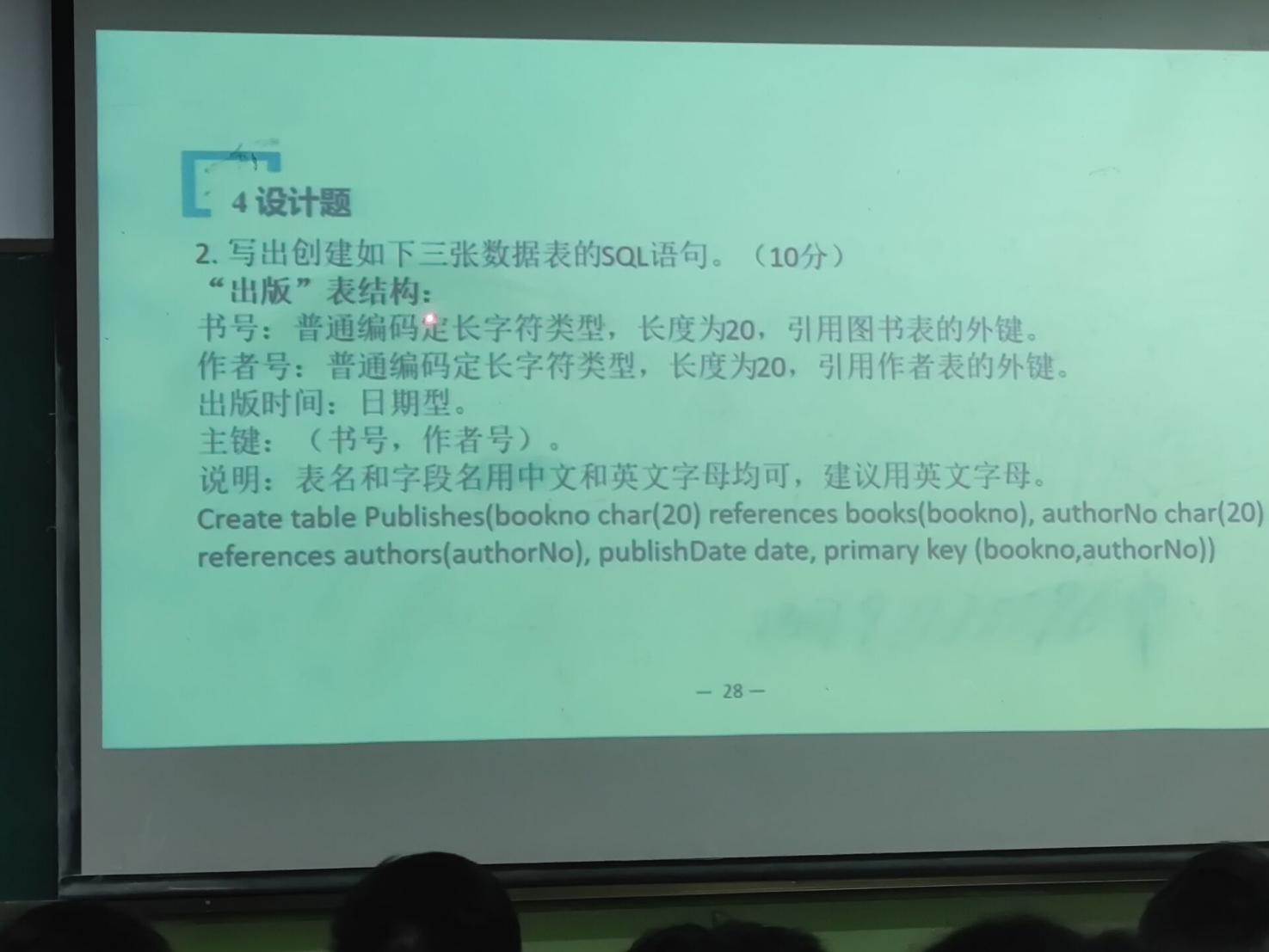
设计题：



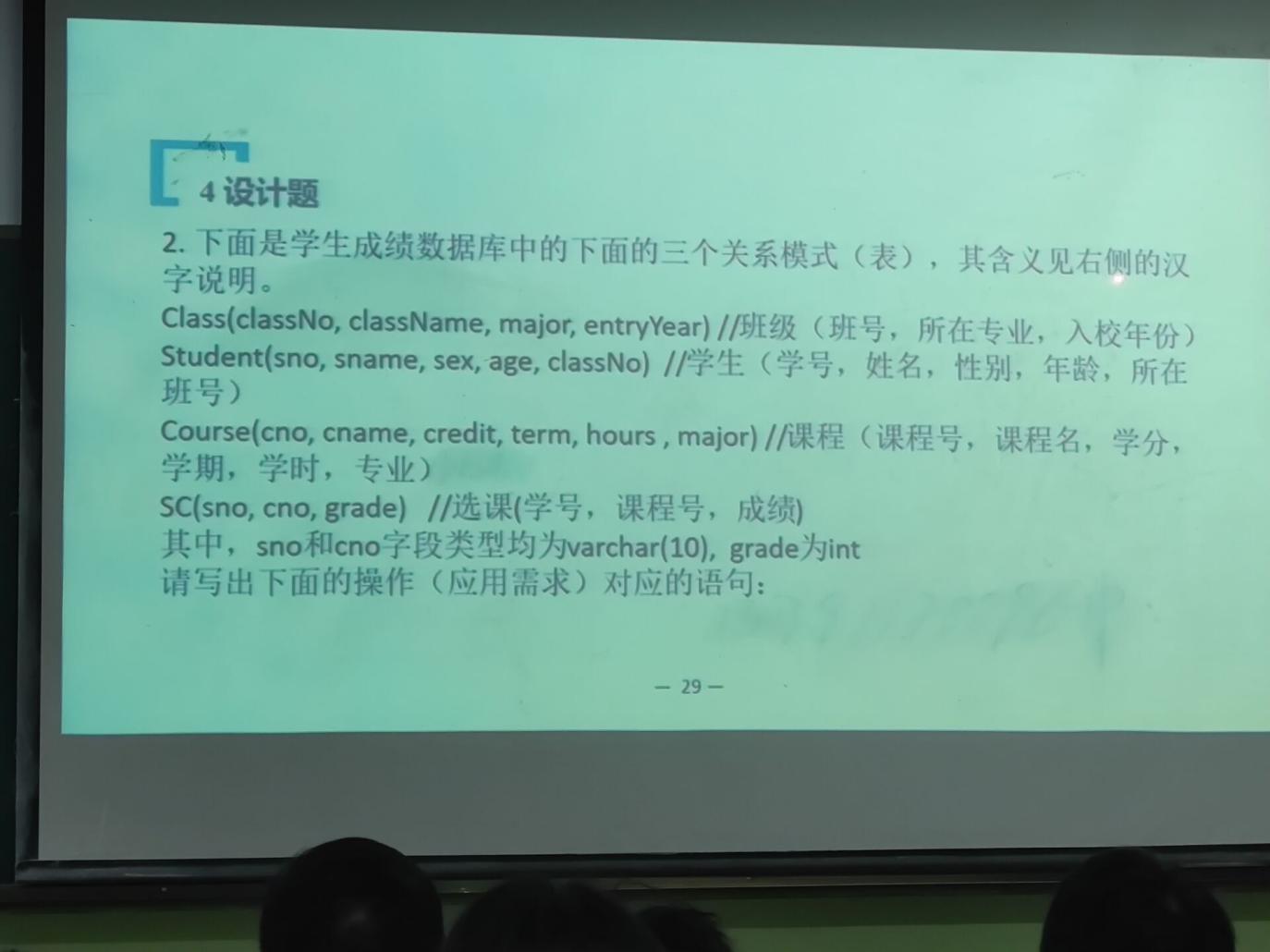




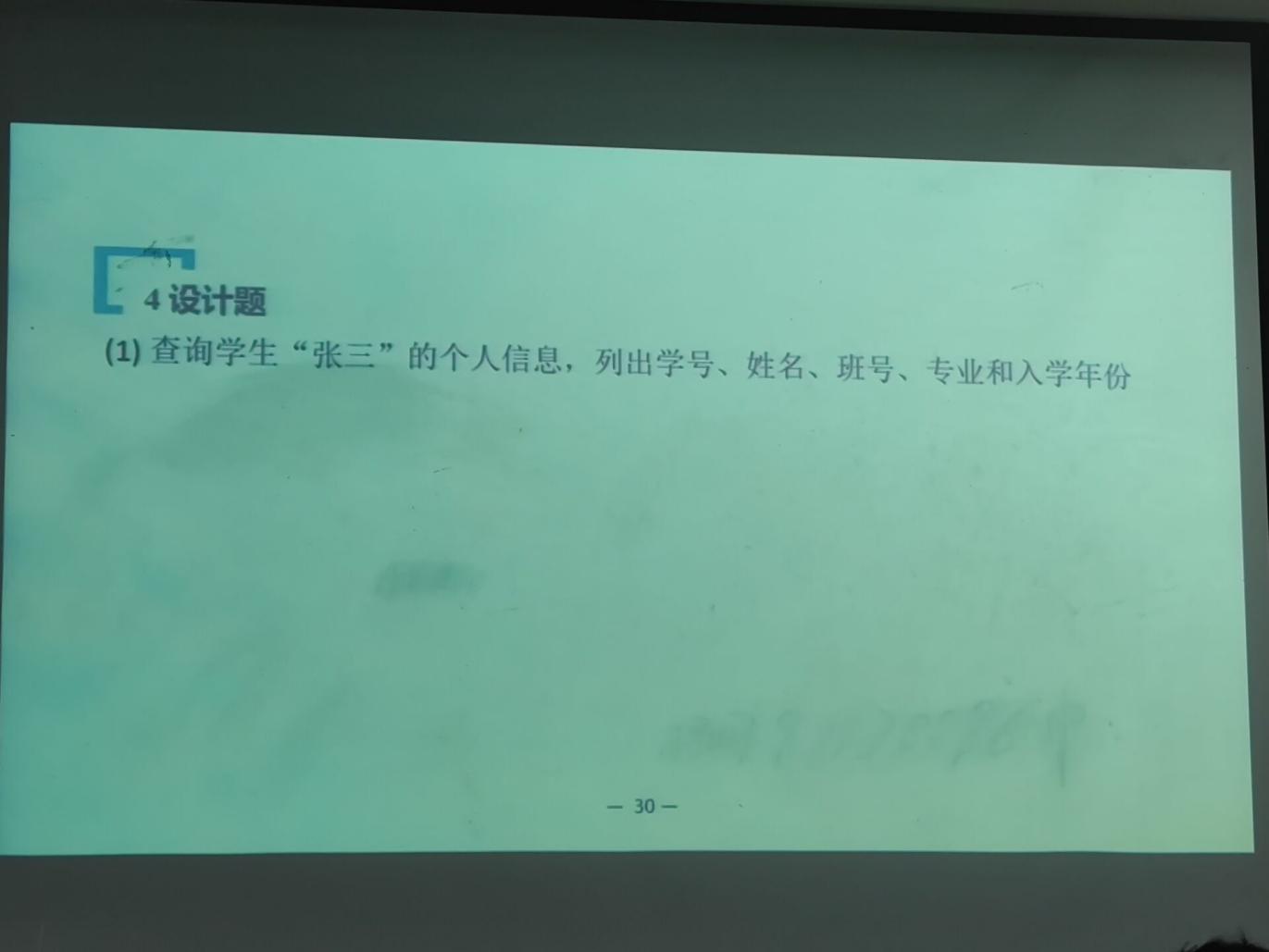
create table authors*(* authorNo char*(*20*)* primary key,  
 authorName varchar*(*40*)* not null,  
 title char*(*4*)  
)*



create table publishes*(* bookNo char*(*20*)* references books*(bookNo)*,  
 authorNo char*(*20*)* references authors*(*authorNo*)*,  
 publishDate date,  
 primary key*(*bookNo,authorNo*)  
)*



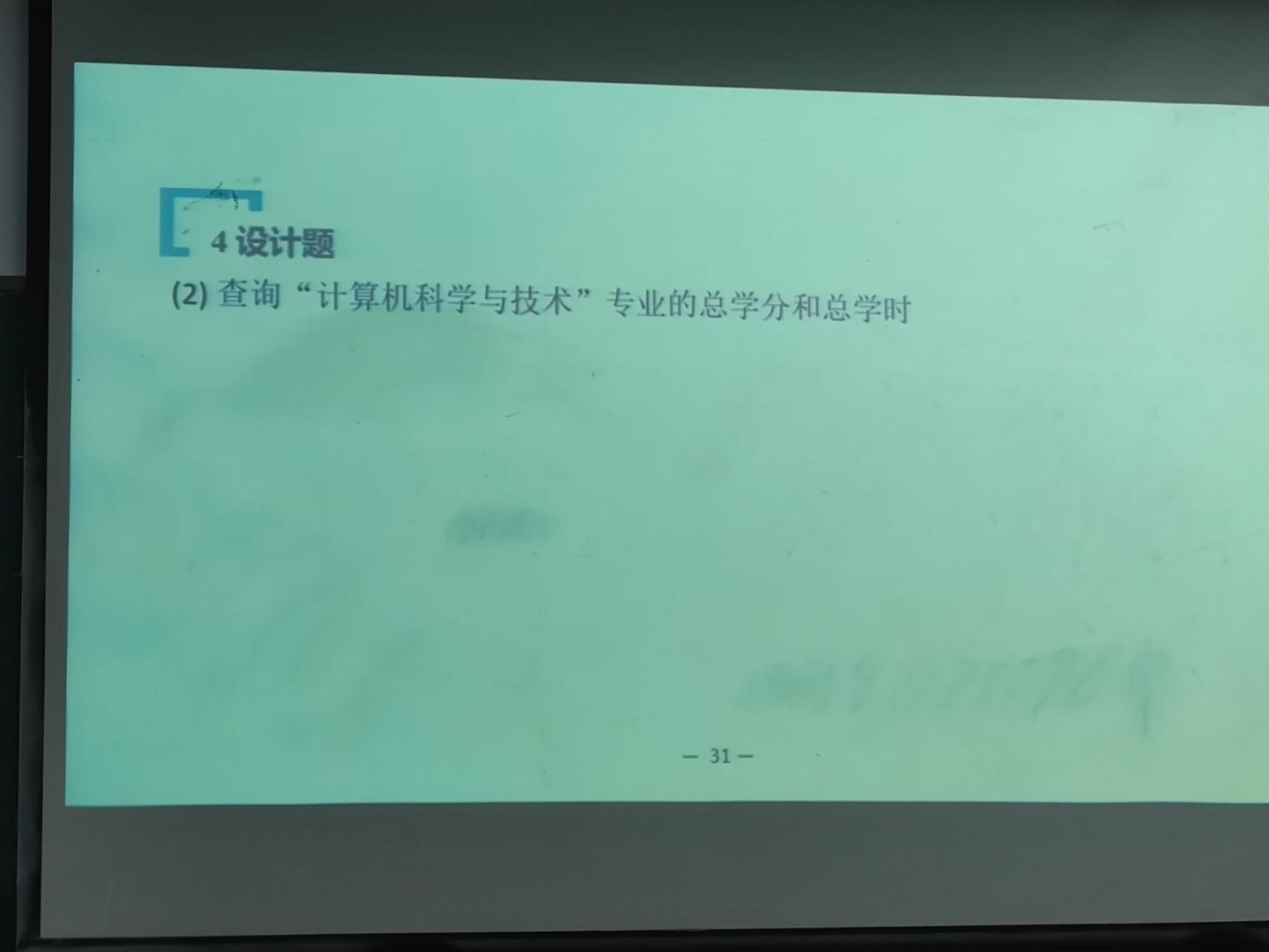
1.



select sno,sname,Student.classNo,major,entryYear  
from Class,Student  
where Class.classNo = Student.classNo  
and sname='张三'

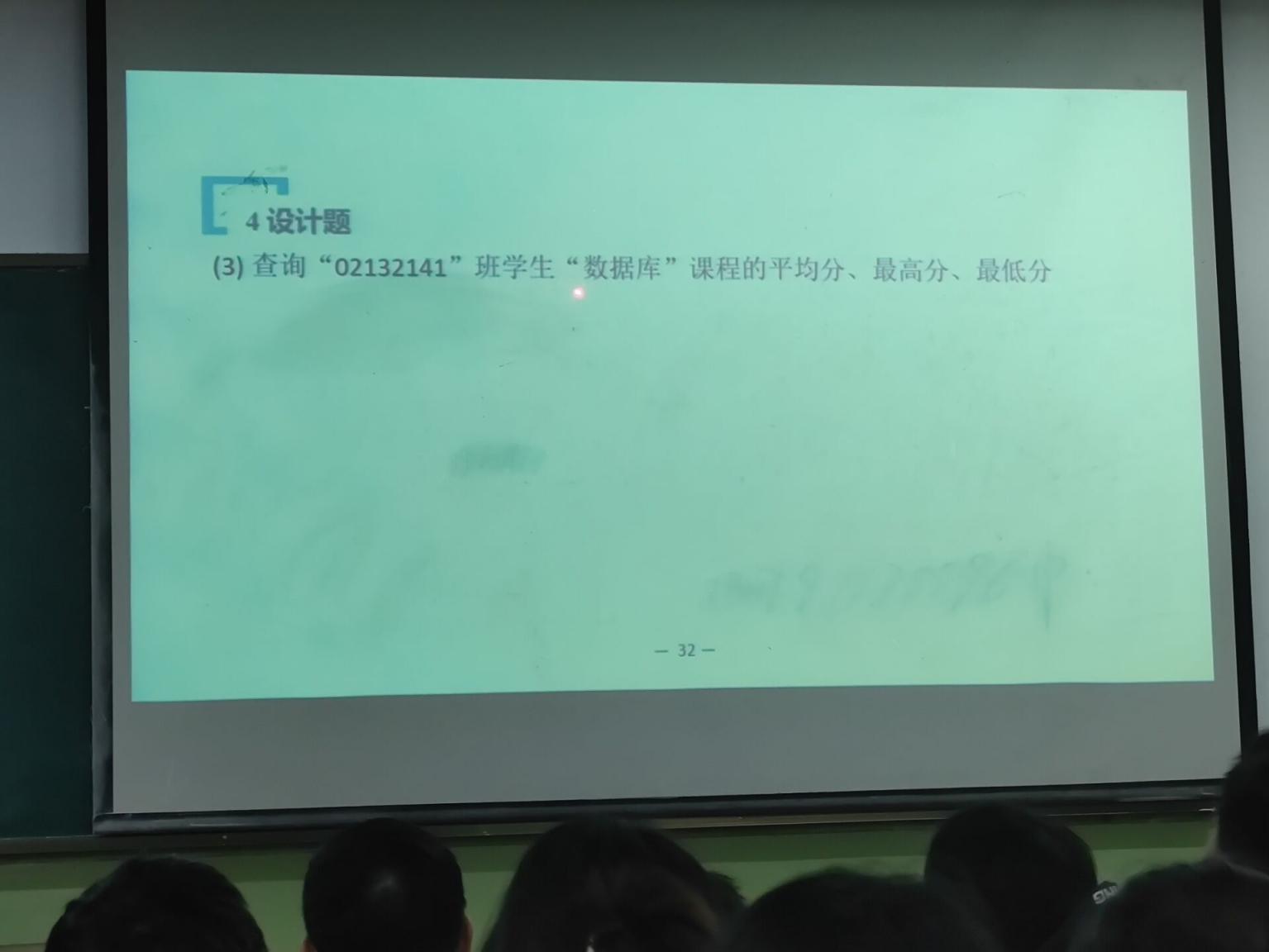
2.

select *sum(credit)* 总学分,*sum(hours)* 总学时  
from Course where *major*='计算机科学与技术'



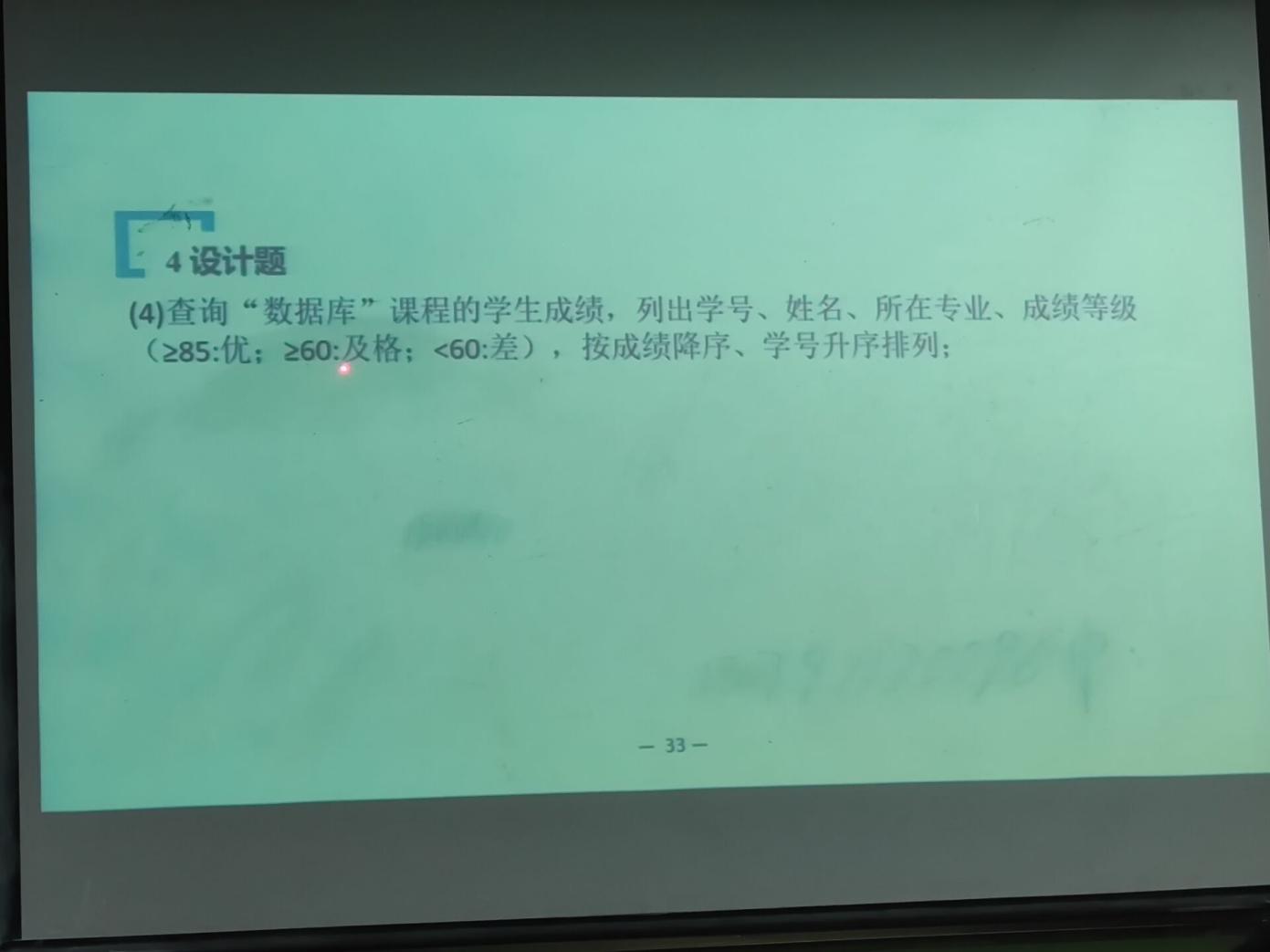
3.

select *avg(grade)*,*max(grade)*,*min(grade)*from SC where *sno* in *(* select *sno* from Student where *classNo*='02132141'  
 *)*



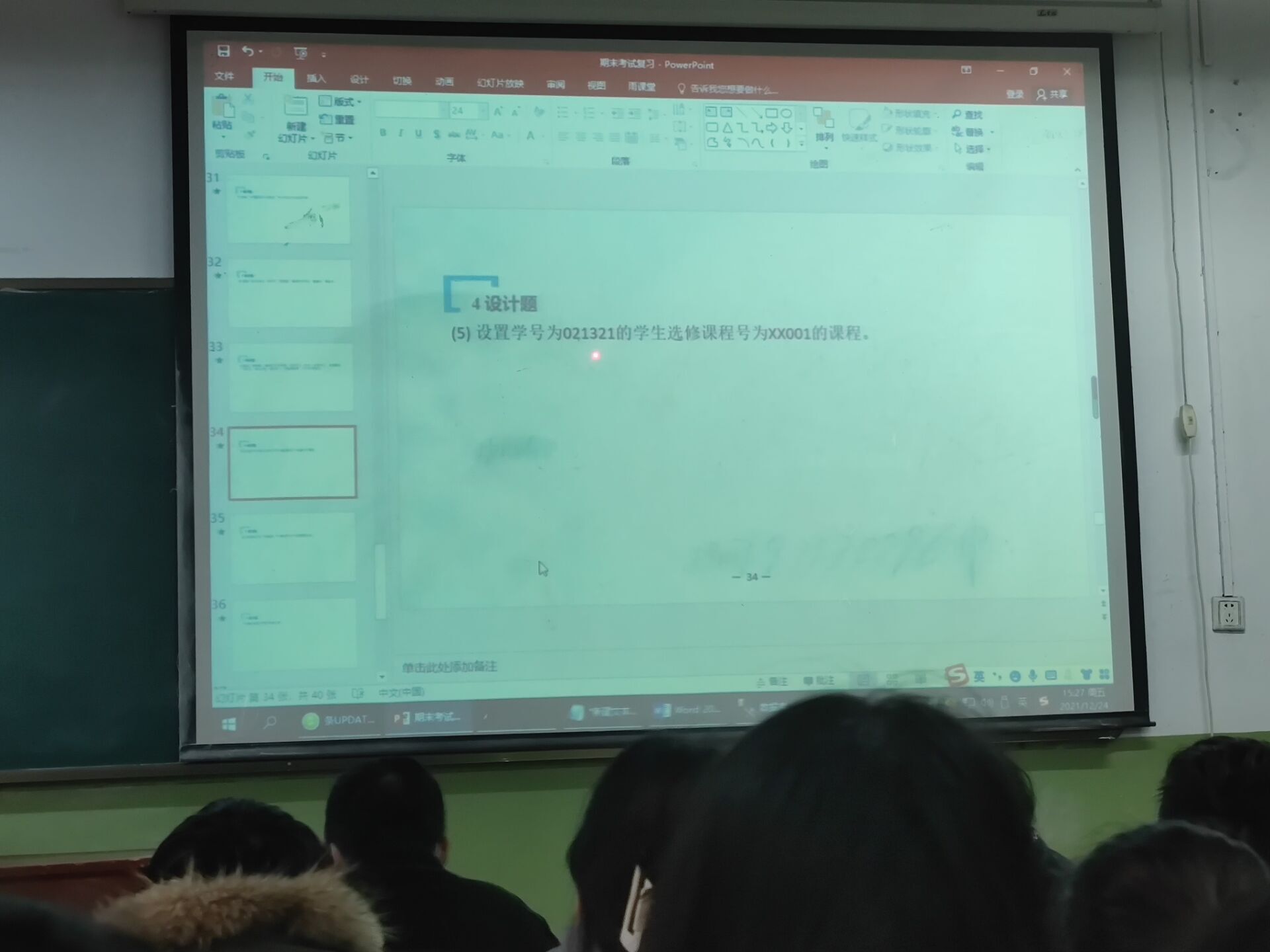
select *avg(isnull(*grade,0*))* 平均分,*max(*grade*)* 最高分,*min(*grade*)* 最低分  
from SC,Course,Student  
where SC.sno=Student.sno and SC.cno=Course.cno  
and classNo='02132141'  
and cname='数据库'

4.



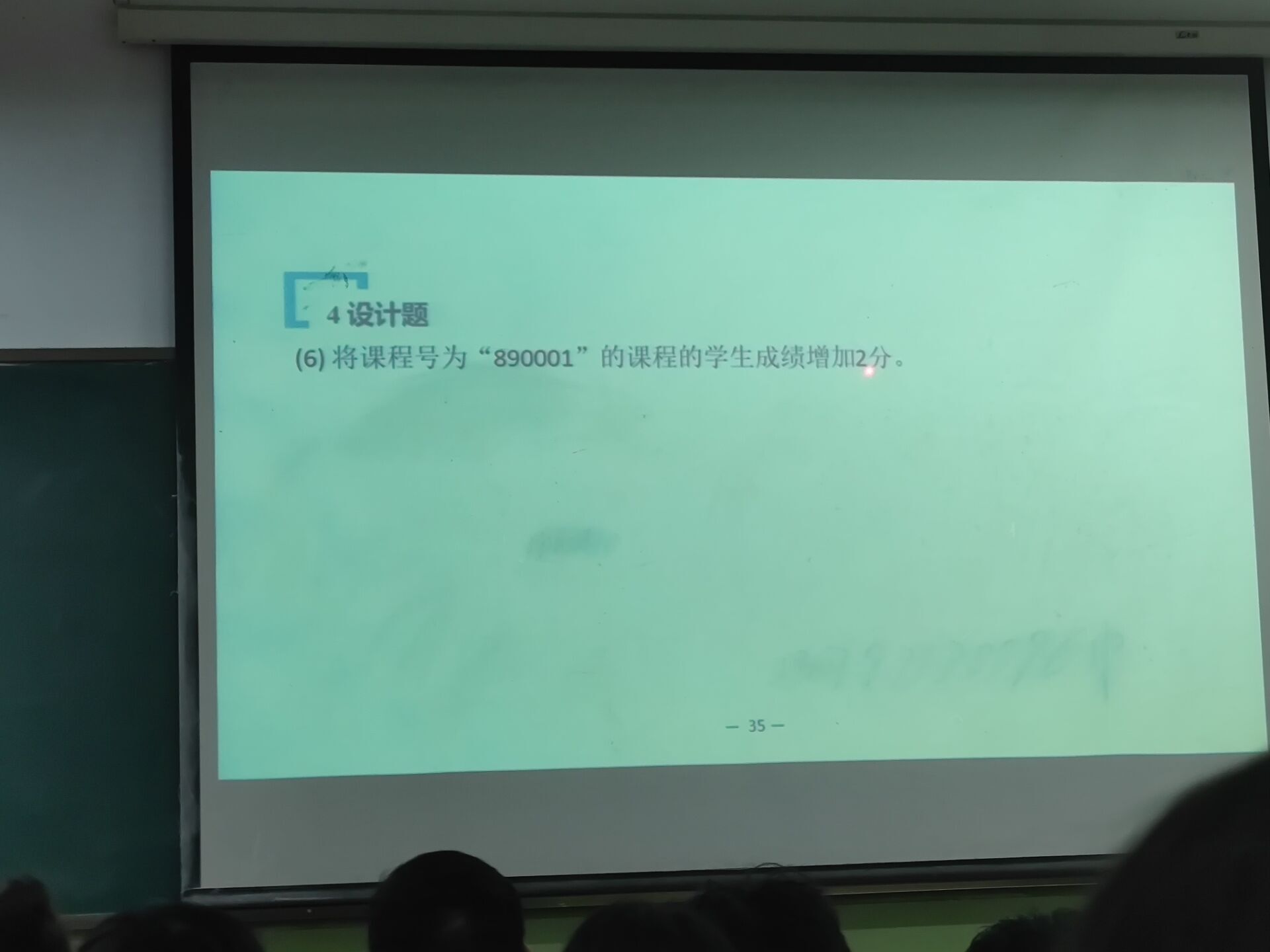
select SC.sno,sname,major,成绩等级 = *case*when grade>=85 then '优'  
when grade>=60 then '及格'  
when grade<60 then '差'  
*end*from SC,Student,Course  
where SC.sno = Student.sno  
and Course.cno = SC.cno  
and cname='数据库'  
order by grade desc,SC.sno asc;

5.



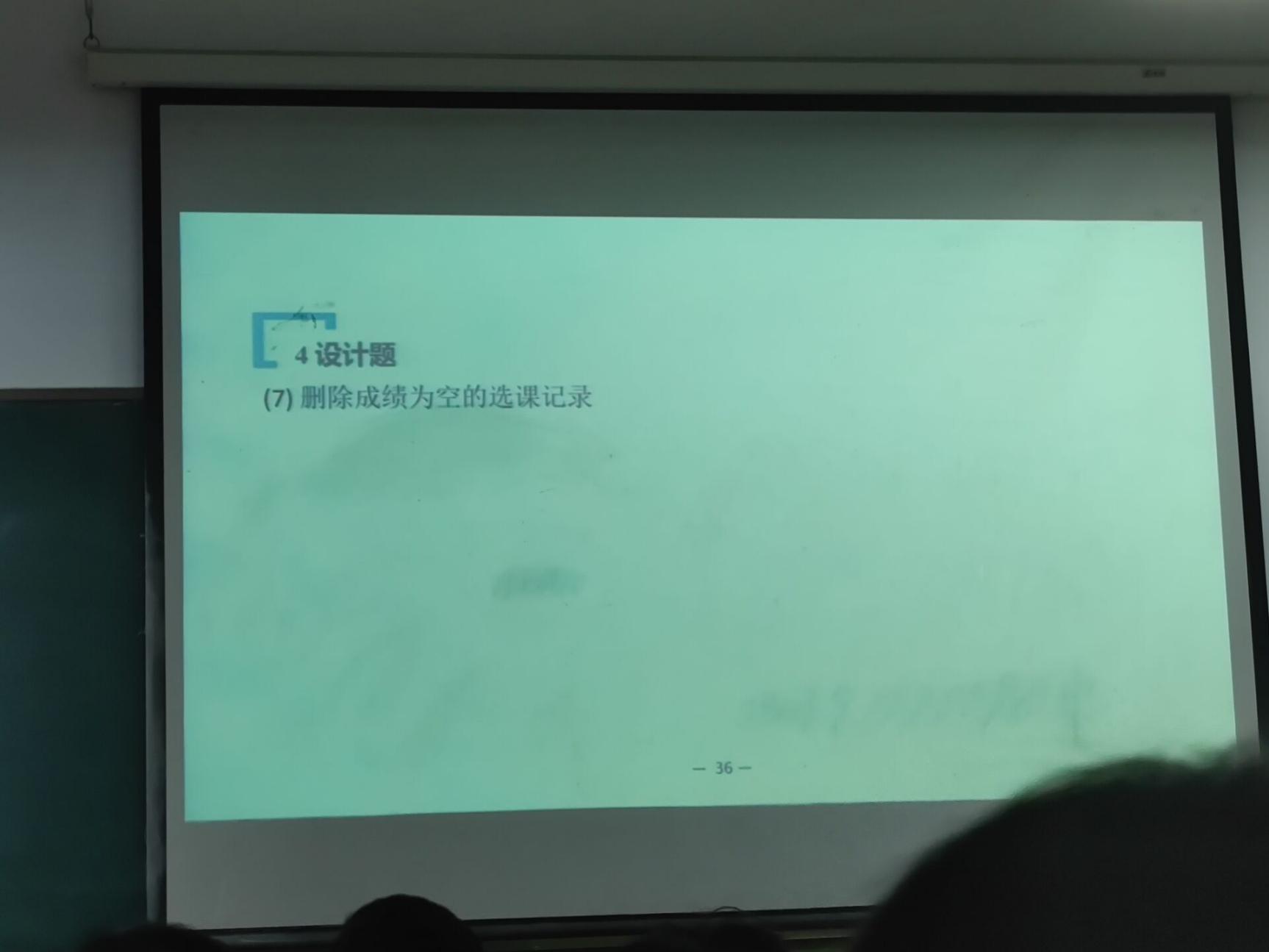
insert into SC*(sno*,*cno)* values *(*'021321','XX001'*)*

6.



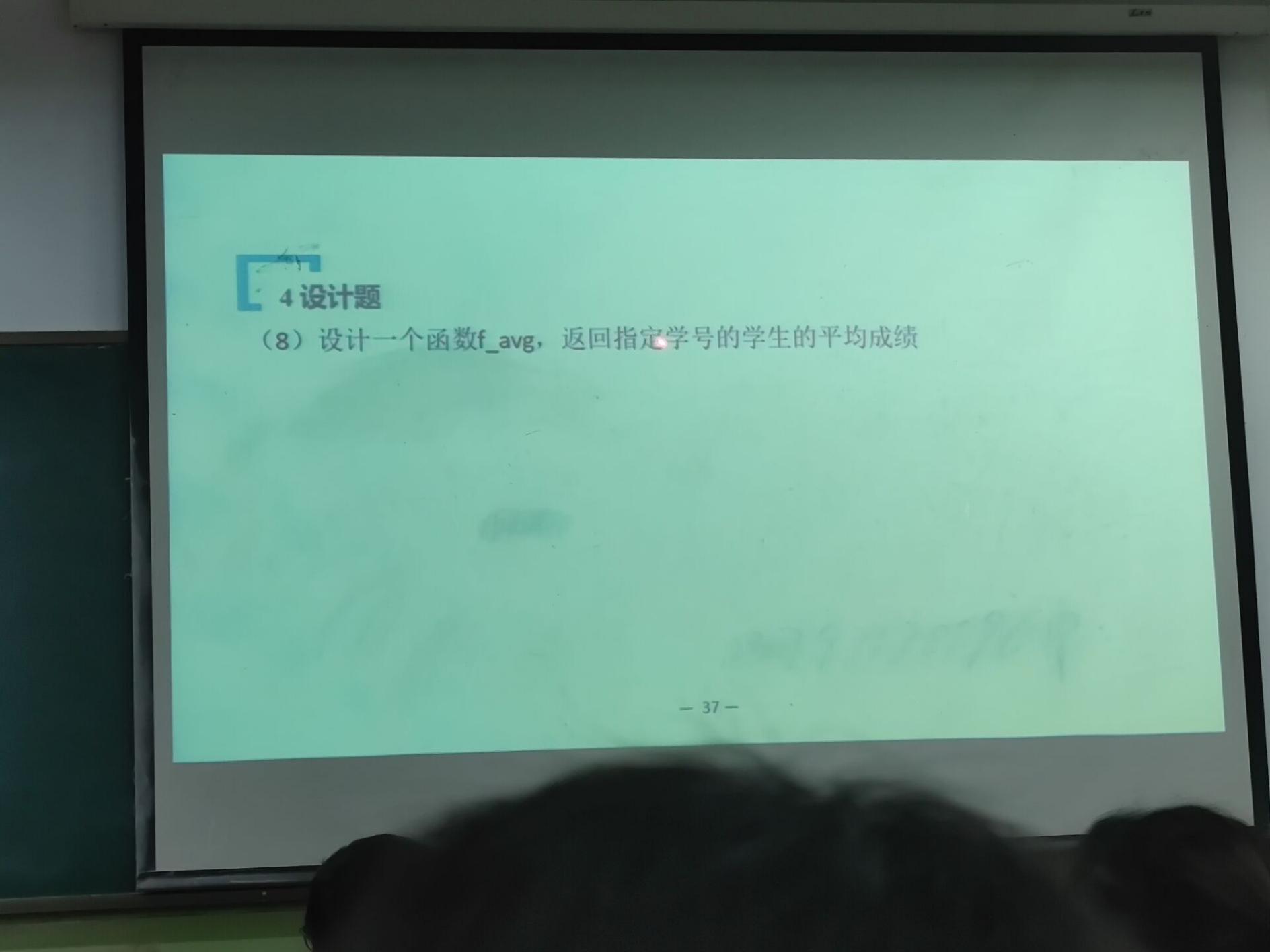
update SC set *grade*=*grade*+2 where *cno*='890001'

7.



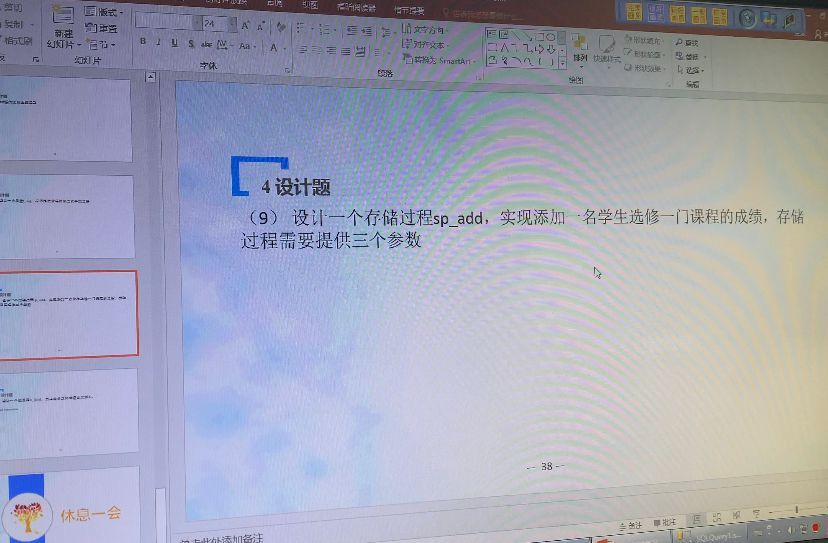
delete from SC where *grade* is null

8.



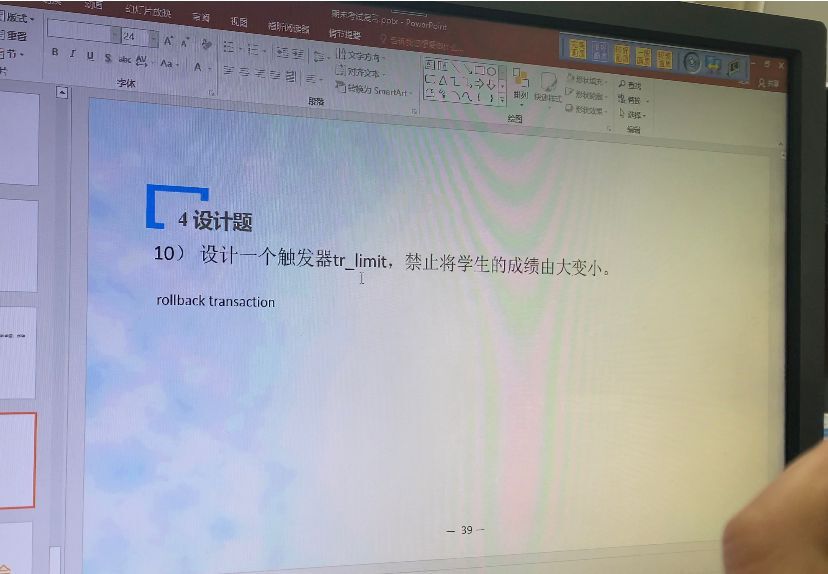
create function *f\_avg(*@sno int*)*returns int  
as  
 *begin* declare @avgGrade int;  
 select @avgGrade=*avg(isnull(grade*,0*))* from SC  
 where *sno*=@sno;  
 return @avgGrade  
 *end*

9.



create procedure *sp\_add(*@sno varchar*(*10*)*,@cno varchar*(*10*)*,@grade int*)*as  
 insert into SC values *(*@sno,@cno,@grade*)*;

10.



create trigger tr\_limit  
 on SC for update  
 as  
*begin* declare @oldGrade int  
 declare @newGrade int  
 declare @sno int  
 select @sno=*sno*,@oldGrade=*grade* from deleted  
 select @newGrade=*grade* from inserted  
 if @oldGrade<@newGrade  
 update SC set *grade*=@newGrade where *sno*=@sno  
 else rollback transaction;  
*end*;

drop trigger tr\_limit;  
  
create trigger tr\_limit  
on cjb for update  
as  
 *begin* declare @oldSco int  
 declare @newSco int  
 declare @id int  
 select @id=学号,@oldSco=成绩 from deleted  
 select @newSco=成绩 from inserted  
 select @oldSco,@newSco,@id;  
 if @oldSco<@newSco  
 update cjb set 成绩=@newSco where 学号=@id  
 else rollback transaction;  
 *end*;  
  
update cjb set 成绩=60 where 学号=191301;