1、存储函数

(1)创建一个存储函数，返回员工的总人数。

(2)创建一个存储函数，删除在 Salary 表中有但在 Employees 表中不存在的员工号。若在 Employees 表中存在则返回 FALSE，若不存在则删除该员工号并返回 TRUE。

(3)创建存储函数，判断员工是否在研发部工作，若是则返回其学历，若不是则返回字符串“NO”。

(4)创建一个存储函数，将工作时间满 4 年的员工收入增加 500 元。

```
use yggl;
    （1）
    drop function if exists e_num;
    delimiter $$
    create function e_num()
    returns int
    begin
        declare num int;
        select count(*) from Employees into num;
        return num;
    end $$
    select e_num();
    （2）
    drop function if exists e_xg;
    delimiter $$
    create function e_xg(id int)
    returns char(6)
    begin
        declare num int;
        declare jieguo char(6);
        select count(*) from salary where id=EmployeeID into num;
        if num=0 then
            delete    from salary where id=EmployeeID;
            set jieguo='True';
        else
            set jieguo='False';
        end if;
        return jieguo ;
    end $$
    select e_xg(000001);

    # 不带参数
    drop function if exists delediff;
    delimiter $$
    create function delediff()
    returns tinyint
```

```
begin
      declare cnt int;
      select count(*) from Salary where EmployeeID not in (
            select EmployeeID from Employees) into cnt;
      if cnt = 0 then
          return false;
      else
          delete from Salary where EmployeeID not in (
                select EmployeeID from Employees);
          return true;
       end if;
end $$

delimiter ;
select delediff();
```

（3）

```
drop function if exists e_judge;
delimiter $$
create function e_judge(e_name char(8))
returns char(12)
begin
      declare id int;
      declare jieguo char(12);
      select DepartmentID from employees where Name=e_name into id;
      if id=4 then select Education from employees where Name=e_name into jieguo;
      else set jieguo='NO';
      end if;
      return jieguo;
end $$
select e_judge('王林');
select e_judge('叶凡');
```

（4）

```
drop function if exists e_salary;
delimiter $$
create function e_salary()
returns int
begin
     update    employees e join salary s on e.EmployeeID=s.EmployeeID
           set InCome=InCome+500 where workyear>=4 ;
     return 0;
end $$
select * from salary;
select e_salary();
```

2.

（1）

```
drop trigger if exists tri_delete;
delimiter $$
create trigger tri_delete
   after delete on employees
   for each row
   begin
      elete from salary where EmployeeID=old.EmployeeID;
   end $$
delete from employees where EmployeeID=000001;
select * from departments2;
create table departments2 as(select * from departments);
drop table departments2;
```

（2）

```
   drop trigger if exists tri_updepartment;
   delimiter $$
   create trigger tri_updepartment
   after insert on departments
   for each row
   begin
         insert into departments2
                  select * from departments where DepartmentID=new.DepartmentID;
   end $$
```

（3）

```
   drop trigger if exists tri_upemployees;
   delimiter $$
   create trigger tri_upemployees
   after update on employees
   for each row
   begin
         declare delta int;
         select new.WorkYear-old.WorkYear from employees
                  where EmployeeID=new.EmployeeID into delta;
         if delta>0 then
                  update    salary set income=income+500*delta
                        where EmployeeID=new.EmployeeID;
         end if;
   end $$
   update employees set WorkYear=4 where Name='伍容华';
   select * from employees where EmployeeID='010008';
```

（4）

写法一：

```
   drop trigger if exists tri_depart;
```

```
delimiter $$
create trigger tri_depart
after update on departments
for each row
begin
        update employees set DepartmentID=new.DepartmentID
            where DepartmentID=old.DepartmentID ;
end $$

delimiter ;
update departments set departmentID=11 where DepartmentName='市场部' ;
```

写法二：

```
drop trigger if exists tri_depart;
delimiter $$
create trigger tri_depart
after update on departments
for each row
begin
declare id1,id2 char(3);
set id1=(select new.departmentid from departments where DepartmentID=new.DepartmentID );
set id2=(select old.departmentid from departments where DepartmentID=new.DepartmentID );
update employees set DepartmentID=id1 where DepartmentID=id2 ;
end $$
delimiter ;
update departments set departmentID=7 where DepartmentName='市场部';
select * from employees;
select * from departments;
```

3.

（1）

```
create event t1
    on schedule at now()
    do
    select * from employees;
    show events;
```

（2）

```
create event t2
    on schedule every 1 day
    starts now()+interval 1 day
    ends '2022-12-31'
    do
```

```
    select * from employees;
（3）
    create event t3
        on schedule at'2022-5-1 11:00:00'
        do
        select * from employees;
（4）
    create event t4
        on schedule every 1 month
        starts date_format(date_add(curdate(),interval 1 month),'%Y-%m-01')
        ends '2022-12-31'
        do
        select * from employees;
```