pl/sql 是什么?
（procedure language/sql）过程化语言，是Oracle在标准的sql上的扩展。

学习的必要性?
1. 提高应用程序的性能
2. 模块化的设计思想
3. 减少网络传输量
4. 提供安全性

缺点?
移植性不好

- 存储过程

例子：

```
SQL> create or replace procedure sp_pro1 is
  2  begin
  3    insert into mytest values ('一梦','m1234');
  4  end;
  5  /

SQL> exec sp_pro1;                //调用存储过程



SQL> create or replace procedure sp_pro2 is
  2  begin
  3    delete from mytest where name = '一梦';
  4  end;
  5  /

SQL> show error;              //查看错误
```

- 块

-- 最简单的块

```
set serveroutput on;        -- 打开输出选项
begin
  dbms_output.put_line('hello world!');
end;

-- 有定义和执行部分的块
declare
v_ename varchar2(5);
v_sal number(7,2);
begin
  select ename,sal into v_ename,v_sal from emp where empno = &no;
  dbms_output.put_line('用户名是:'||v_ename || '，工资：' || v_sal);
end;

-- 异常处理
declare
v_ename varchar2(5);
v_sal number(7,2);
begin
  select ename,sal into v_ename,v_sal from emp where empno = &no;
  dbms_output.put_line('用户名是:'||v_ename || '，工资：' || v_sal);
  exception
    when no_data_found then
      dbms_output.put_line('没有查到对应的数据！');
end;
```

- 存储过程

修改scott用户的薪水?

```
SQL> create or replace procedure sp_pro3(spName varchar,newSal number) is
  2  begin
  3    update emp set sal=newSal where ename = spName;
  4  end;
  5  /
```

```
SQL>exec sp_pro3('SCOTT',9999);
```

- 函数

```
SQL> create or replace function sp_fun1(spName varchar2) return number is yearSal number(11,2);
  2  begin
  3    select sal*12+nvl(comm,0)*12 into yearSal from emp  where ename = spName;
  4    return yearSal;
  5  end;
  6  /
SQL> var a number;
SQL> call sp_fun1('SCOTT') into:a;
```

- 包

创建包

```
SQL> create or replace package sp_pk1 is
  2  procedure update_sal(name varchar2,newsal number);
  3  function annual_income(name varchar2) return number;
  4  end;
  5  /
```

创建包体

```
SQL> create or replace package body sp_pk1 is
  2  procedure update_sal(name varchar2,newsal number)
  3    is
  4    begin
  5      update emp set sal=newsal where ename=name;
  6    end;
  7  function annual_income(name varchar2)
  8    return number is
  9    annual_salary number;
 10    begin
 11      select sal*12+nvl(comm,0)*12 into annual_salary from emp where
ename=name;
 12      return annual_salary;
 13      end;
 14  end;
 15  /
```

调用包过程和函数

```
SQL> exec sp_pk1.update_sal('SCOTT','0');
SQL> call sp_pk1.annual_income('SCOTT') into:a;
```

-- 输入员工工号，显示员工姓名、工资、个人所得税（税率0.03）？

```
declare
v_rate number(3,2):=0.03;
v_ename emp.ename%type;
v_sal number(7,2);
v_tax_sal number(7,2);
begin
     select ename,sal into v_ename,v_sal from emp where empno=&no;
     v_tax_sal:=v_rate*v_sal;
     dbms_output.put_line('员工姓名：'|| v_ename || ' 工资：' || v_sal || ' 个人所得税：' ||
v_tax_sal);
end;
```

游标

```
SQL> declare
  2  -- 定义游标类型sp_emp_cursor
  3  type sp_emp_cursor is ref cursor;
  4  --定义一个游标变量
  5  test_cursor sp_emp_cursor;
  6  --定义变量
  7  v_ename emp.ename%type;
  8  v_sal emp.sal%type;
  9  begin
 10    open test_cursor for select ename,sal from emp where deptno=&no;
 11    --循环取出test_cursor
 12    loop
 13      fetch test_cursor into v_ename,v_sal;
 14      -- 判定test_cursor为空
 15      exit when test_cursor%notfound;
 16      dbms_output.put_line('名字：'||v_ename || '工资： ' || v_sal);
 17    end loop;
 18  close test_cursor;
 19  end;
```

```
end;
/
```

案例：

--编写一个存储过程，输入一个雇员名，如果工资低于2000就给给雇员工资增加10%

```
create or replace procedure sp_update_sal(spName varchar) is
v_sal emp.sal%type;
begin
  select sal into v_sal from emp where ename = spName;
  if v_sal<2000 then
    update emp set sal=sal+sal*0.1 where ename=spName;
  end if;
end;
```

--编写一个存储过程，输入一个雇员名，如果补助不是0就在原基础上加100，如果补助是0就增加200

```
create or replace procedure sp_update_comm(spName varchar) is
v_comm emp.comm%type;
begin
  select comm into v_comm from emp where ename = spName;
  if v_comm<>0 then
    update emp set comm=comm+100 where ename=spName;
  else
    update emp set comm=comm+200 where ename=spName;
  end if;
end;
```

--编写一个存储过程，输入一个员工编号，如果该员工职位是 PRESIDENT 就给工资增加1000 ，如果是 MANAGER 就给工资增加500，其它职位员工增加200

```
create or replace procedure sp_update_newsal(spno number) is
v_job emp.job%type;
begin
  select job into v_job from emp where empno = spno;
  if v_job='PRESIDENT' then
    update emp set sal=sal+1000 where empno=spno;
  elsif v_job='MANAGER' then
    update emp set sal=sal+500 where empno=spno;
  else
     update emp set sal=sal+200 where empno=spno;
  end if;
end;
```

循环

loop

--编写一个过程，输入一个用户名，循环添加10次，用户id从1递增

```
create table users1 (usersID number,usersName varchar2(15));

create or replace procedure sp_pro6(spName varchar2) is
v_num number:=1;
begin
  loop
    insert into users1 values(v_num,spName);
    exit when v_num=10;
    v_num:=v_num+1;
  end loop;
end;
```

while